COMPUTER-AIDED DESIGN,
ENGINEERING, AND MANUFACTURING
SYSTEMS TECHNIQUES AND APPLICATIONS

VOLUME
VI

MANUFACTURING SYSTEMS PROCESSES

COMPUTER-AIDED DESIGN,
ENGINEERING, AND MANUFACTURING
SYSTEMS TECHNIQUES AND APPLICATIONS

VOLUME
VI

MANUFACTURING SYSTEMS PROCESSES

EDITOR
CORNELIUS LEONDES

CRC Press
Boca Raton London New York Washington, D.C.

Library of Congress Cataloging-in-Publication Data

Catalog record is available from the Library of Congress.

This book contains information obtained from authentic and highly regarded sources. Reprinted material is quoted with permission, and sources are indicated. A wide variety of references are listed. Reasonable efforts have been made to publish reliable data and information, but the author and the publisher cannot assume responsibility for the validity of all materials or for the consequences of their use.

Neither this book nor any part may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, microfilming, and recording, or by any information storage or retrieval system, without prior permission in writing from the publisher.

The consent of CRC Press LLC does not extend to copying for general distribution, for promotion, for creating new works, or for resale. Specific permission must be obtained in writing from CRC Press LLC for such copying.

Direct all inquiries to CRC Press LLC, 2000 N.W. Corporate Blvd., Boca Raton, Florida 33431.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation, without intent to infringe.

© 2001 by CRC Press LLC

No claim to original U.S. Government works
International Standard Book Number 0-8493-0998-0
Printed in the United States of America 1 2 3 4 5 6 7 8 9 0
Printed on acid-free paper

Preface

A strong trend today is toward the fullest feasible integration of all elements of manufacturing including maintenance, reliability, supportability, the competitive environment, and other areas. This trend toward total integration is called concurrent engineering. Because of the central role information processing technology plays in this, the computer has also been identified and treated as a central and most essential issue. These are the issues that are at the core of this volume.

This set of volumes consists of seven distinctly titled and well-integrated volumes on the broadly significant subject of computer-aided design, engineering, and manufacturing: systems techniques and applications. It is appropriate to mention that each of the seven volumes can be utilized individually. In any event, the great breadth of the field certainly suggests the requirement for seven distinctly titled and well integrated volumes for an adequately comprehensive treatment. The seven volume titles are:

1. Systems Techniques and Computational Methods
2. Computer-Integrated Manufacturing
3. Operational Methods in Computer-Aided Design
4. Optimization Methods for Manufacturing
5. The Design of Manufacturing Systems
6. Manufacturing Systems Processes
7. Artificial Intelligence and Robotics in Manufacturing

The contributions to this volume clearly reveal the effectiveness and great significance of the techniques available and, with further development, the essential role they will play in the future. I hope that practitioners, research workers, students, computer scientists, and others on the international scene will find this set of volumes to be a unique and significant reference source for years to come.

Cornelius T. Leondes
Editor

© 2001 by CRC Press LLC

Editor

Cornelius T. Leondes, B.S., M.S., Ph.D. Emeritus Professor, School of Engineering and Applied Science, University of California, Los Angeles. He has served as a member or consultant on numerous national technical and scientific advisory boards. He has served as a consultant for numerous Fortune 500 companies and international corporations. Has published more than 200 technical journal articles and has edited and/or co-authored more than 120 books. He is a Guggenheim Fellow, Fullbright Research Scholar, and Fellow of IEEE, recipient of the IEEE Baker Prize award, and recipient of the Barry Carlton Award of the IEEE.

© 2001 by CRC Press LLC

Contributors

Tudor Balan
CER ENSAM de Metz
Metz, France

Raj Balendra
University of Strathclyde
Glasgow, United Kingdom

Robert Joel Barnett
Vanderbilt University
Nashville, Tennessee

Chie-Bein Chen
Ming-Chuan University
Taipei, Taiwan

Jean-Loup Chenot
Center for Forming Processes
Antipolis, France

George E. Cook
Vanderbilt University
Nashville, Tennessee

Richard C. Dorf
University of California at Davis
Davis, California

Lionel Fourment
Center for Forming Processes
Antipolis, France

Daniel A. Hartman
Los Alamos National Laboratory
Los Alamos, New Mexico

Hodge E. Jenkins
Lucent Technologies, Inc.
Norcross, Georgia

Thomas R. Kurfess
Georgia Institute of Technology
Atlanta, Georgia

S.G. Lambrakos
Naval Research Laboratory
Washington, D.C.

D.W. Moon
Naval Research Laboratory
Washington, D.C.

Yi Qin
University of Strathclyde
Glasgow, United Kingdom

R.S. Srinivasan
The University of Texas
Austin, Texas

Alvin M. Strauss
Vanderbilt University
Nashville, Tennessee

Chih-Hung Tsai
Ta-Hwa Institute of Technology
Hsin-Chu, Taiwan

Irem Y. Tumer
NASA Ames Research Center
Moffett Field, California

Kristin L. Wood
The University of Texas
Austin, Texas

Chiu-Chi Wei
Chung-Hua University
Hsin-Chu, Taiwan

© 2001 by CRC Press LLC

Contents

Chapter 1 Analysis and Synthesis of Engineering Surfaces in Bridge Manufacturing and Design
Irem Y. Tumer, R. S. Srinivasan, and Kristin L. Wood

Chapter 2 Determination of Achievable Process Tolerances with Respect to Manufacturing Process Capability
Chiu-Chi Wei, Chih-Hung Tsai, and Chie-Bein Chen

Chapter 3 Production of High Quality Parts by the Process of Grinding in Manufacturing Systems
Thomas R. Kurfess, Hodge E. Jenkins, and Richard C. Dorf

Chapter 4 Finite Element Modeling and CAD/CAM of Nett-Forming by Injection Forging
Raj Balendra and Yi Qin

Chapter 5 Optimal Design Techniques in Non-Steady-State Metal Forming Processes
Lionel Fourment, Tudor Balan, and Jean-Loup Chenot

Chapter 6 Analysis of Welds Using Geometric Constraints
S.G. Lambrakos and D.W. Moon

Chapter 7 Neural Network Systems Techniques in Weld Modeling and Control
George E. Cook, Robert Joel Barnett, Daniel A. Hartman, and Alvin M. Strauss

© 2001 by CRC Press LLC

1

Analysis and Synthesis of Engineering Surfaces in Bridge Manufacturing and Design

1.1 Introduction
Analysis and Synthesis for Advanced Manufacturing • Relation to Precision Surfaces • A Vision for "Fingerprinting" Manufactured Surfaces
1.2 The Never-Ending Search for the Perfect Surface Analysis Method
Traditional Surface Characterization Methods • Merging Random Process Analysis with Surface Analysis • Criteria for Accurate Signal Analysis and Synthesis • Avenues for Improvement
1.3 Fractal-Wavelet Method
Literature Background: Fractals • Literature Background: Wavelets • Geometric Notion of Fractal Dimensions • Philosophy: Why Fractals in Manufacturing? • Fractal-Wavelet Analysis • Fractal-Wavelet Synthesis • A Superposition Approach • Milling Profiles • Grinding Profiles • Comments on the Analysis • Synthesis and Comparison • Performance Comparison: Axial Vibrations in Ball Bearings
1.4 Karhunen-Loève Method
Philosophy • Literature Background • Mathematical Background • Karhunen-Loève Transform for Surface Analysis • Decomposing Complex Signals into Orthogonal Functions • Monitoring Decomposed Functions in Signals • Predicting Fault Patterns in Signals • Monitoring and Prediction of Numerically Generated Surface Signals • Understanding and Redesigning a Manufacturing Process through Surface Analysis
1.5 Discussion: Extensions and Ending Thoughts
Dichotomy or Integration? • Conjunction of the Two Methods • Future Implications • Acknowledgments

Irem Y. Tumer
NASA Ames Research Center

R.S. Srinivasan
The University of Texas

Kristin L. Wood
The University of Texas

1.1 Introduction

Today's advanced manufacturing and design requirements necessitate tools to improve the quality of parts and assemblies. Quality issues range from form and size deviations to surface irregularities, deviations in strength and material properties, and feature misplacement. In this chapter, we concentrate on the class of quality issues associated with form and surface deviations. Currently, manufacturing lacks robust and accurate methods to analyze and predict the nature of surfaces generated from manufacturing processes. Designers lack accurate manufacturing information about the status of the product. Accurate information about the fault status during manufacturing is necessary for designers to take remedial actions such as redesign of machine components and process parameters. A fault, for our purposes, is defined as the inability of a system to perform in an acceptable manner [55]. Faults typically manifest themselves as deviations in observed behavior from a set of acceptable behaviors. The identification and elimination of the fault mechanisms are essential in assuring the production of precision components in manufacturing. In our work, we aim to develop novel methods to improve the accuracy of surface condition information for manufactured components.

Analysis and Synthesis for Advanced Manufacturing

Advancements in automated manufacturing and design of quality components require a thorough understanding of the precision of products from manufacturing. Advanced manufacturing and design goals require two tasks: (1) the automated detection of fault patterns on manufactured-part surfaces to enable either the redesign of the manufacturing machine or the choice of more appropriate process or design parameters; and (2) the accurate reconstruction of the expected surface patterns to enable accurate prediction of design performance. These tasks can only be accomplished by providing accurate mathematical means of analyzing and synthesizing surface patterns. In this work, surface patterns are used to provide a "fingerprint" of the fault mechanisms in the manufacturing process. Analysis of the surface patterns in this context refers to the decomposition and quantification of the surface characteristics. Analysis will enable the detection and monitoring of significant faults in the system. Synthesis of the surface patterns refers to the inverse process of predicting the surface characteristics, given the manufacturing conditions. The synthesized surfaces can then be used during design to predict product performance.

Relation to Precision Surfaces

The motivation of this work stems from a need to understand and predict the mechanisms which account for the formation of high-precision engineering surfaces [67]. The main objective of efforts in precision engineering is to design machines with high and predictable work-zone accuracies [39]. The end goal is to assure the consistent production of precision surfaces. These efforts lead to the need to understand the factors that affect machine performance, as well as the basic physics that characterizes a machine component or system. This understanding is an essential step for the development of superior designs and proper selection of machine components. The design of quality precision machines depends primarily on the ability of engineers to predict machine performance and monitor process variations [39, 63]. In our work, we attempt to reach this understanding by studying the characteristics of engineering surfaces. Specifically, we believe in the importance of understanding and representing engineering surfaces by means of mathematical tools.

The need for and importance of understanding and characterizing engineering surfaces is portrayed by an analogy to the growth of a tree, as shown in Fig. 1.1. The genesis of any surface is in the manufacturing domain; this is likened to the root of a tree, which initiates growth. There is a hierarchy of several mechanisms that lead to surface generation [57], and they can be attributed to material, process, environment, operator, tooling, etc. In most cases these mechanisms are not apparent and/or well understood, much like the secondary (and tertiary) roots that run deep below the earth, and are hidden

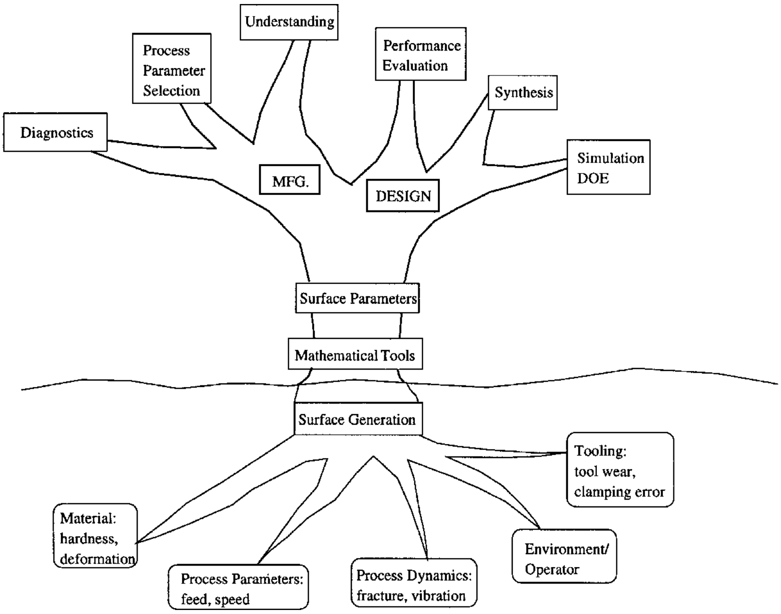

Understanding
Process Parameter Selection
Performance Evaluation
Synthesis
Diagnostics
MFG.
DESIGN
Simulation DOE
Surface Parameters
Mathematical Tools
Surface Generation
Tooling: tool wear, clamping error
Material: hardness, deformation
Environment/ Operator
Process Parameters: feed, speed
Process Dynamics: fracture, vibration

FIGURE 1.1 Importance of mathematical methods for engineering surfaces: tree analogy.

from sight. However, the combined effects of all these factors lead to the final surface, just as nutrients from all the roots lead to the primary root. The continuum of the primary root emerges from the ground as the trunk of the tree, leading to branches and leaves. We equate the branches to design and (the cognition of) manufacturing, as these are the tangible applications of manufactured surfaces, with well established fields of study.

However, in order to understand and control surface generation, we need methods closer to the root. We compare the trunk of the tree to the mathematical tools required to analyze and synthesize surfaces, and also the surface parameters thus derived. Analogous to the rise of nutriments from the root to the leaves via the trunk, mathematical techniques and parameters form the interface between the physicality of surface generation and the sublime, creative realms of design. They equip the designer with the means to take into account the effects of manufactured surfaces on the performance of their artifact. In addition to design, surface parameters can be used for understanding the manufacturing process, selecting process parameters, monitoring, diagnosing, and correcting faults, etc. Through this simile, we underscore the significance of mathematical techniques and parameters to link the domains of manufacturing and design, just as the trunk of a tree connects (and sustains) the roots and branches.

A Vision for "Fingerprinting" Manufactured Surfaces

In our work, we search for accurate mathematical techniques to understand and represent the surface generation process. Methods to analyze surfaces of manufactured parts have existed in many forms, from simple average surface measures, to advanced Fourier-based visualization of the surface information [74]. Researchers have explored many different approaches to quantify surface information. However, shortcomings of the current methods, in the face of a rapidly-advancing manufacturing technology, result in a constant search for a means to revolutionize manufacturing precision. Specifically, a means to fingerprint the manufacturing process is necessary to detect and characterize all significant patterns on product surfaces, isolate the originating source, and identify the corresponding design/redesign problem for the manufacturing machine and/or process. A vision of this revolution is needed as a first step. Our vision builds on contemporary methods, but with three distinct departures: (1) creating a unified approach to

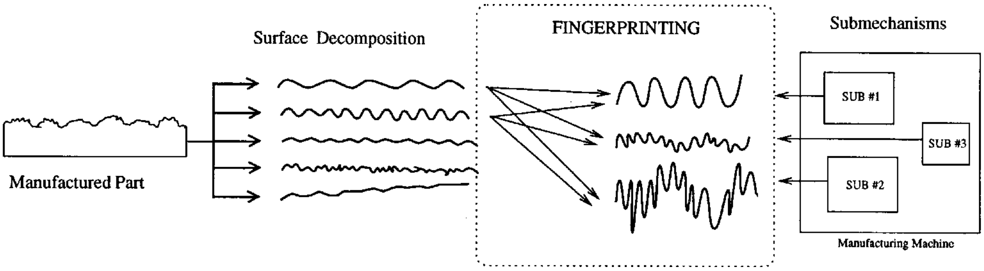


FIGURE 1.2 A vision for manufacturing fingerprinting.

surface analysis and synthesis; (2) representing structure in the stochastic nature of manufacturing processes; and (3) clearly separating the deterministic aspects in the presence of stochastic and time-varying effects. A simple implementation of this vision is shown in Fig. 1.2: surface measurements are decomposed and fingerprinted for the purpose of isolating the faulty submechanisms in the manufacturing system. The remainder of this chapter expounds on this vision through the development of practical tools to understand surface characteristics at a new level. These tools are illustrated with mathematical representations, numerical examples, and industrial applications.

1.2 The Never-Ending Search for the Perfect Surface Analysis Method

Understanding manufactured surfaces has been an interesting research problem for a long time. Surface characterization is meant to decompose the surface geometry into its basic components in order to understand it better [74]. The progressive understanding gained from the numerous different techniques to characterize surfaces has been valuable [74]. However, while the use of some of these techniques has been standardized successfully, the need to develop better and more accurate methods has not ceased. In fact, the growing need for automated manufacturing and design necessitates a revolutionary technique, which will provide a unified means of characterizing all significant components simultaneously. In this section, we present an overview of the traditional techniques developed for surface analysis purposes. We then present some alternative ways of looking at surfaces and our criteria in helping to advance the field of surface analysis.

Traditional Surface Characterization Methods

Manufactured surfaces are typically viewed as containing various types of deviations from the intended nominal surface profile [74]. The widely recognized types of deviations are identified as: (1) irregularities known as surface "roughness" that often result from the manufacturing process, such as tool marks left as a result of turning or marks resulting from a grinding wheel; (2) irregularities known as surface "waviness" of a longer wavelength caused by improper manufacturing, such as vibration between a workpiece and a cutting tool; and (3) low-frequency waveforms or linear trends referred to as errors of "form," such as errors in rotating members of a machine or thermal distortion errors.

The most traditional means of characterizing these various types of surface errors is by using average profile parameters [74]. Two primary parameters are roughness and waviness height measures. The height parameters are typically specified on part models in an attempt to control the manufacturing process. However, efforts to optimize the functionality of a part leads to a need to supplement the height information. One such supplement parameter is the spacing parameter, defined as the average distance between the positive crossings of the profile with the mean line [44, 74]. Other parameters define the maximum range of the profile, as well as the skewness (asymmetry) and the randomness of the profiles.

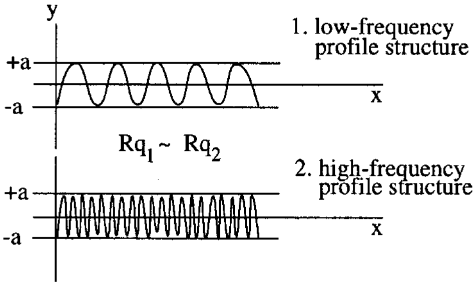

y
+a
-a
1. low-frequency profile structure
x
$Rq_1 \sim Rq_2$
2. high-frequency profile structure
+a
-a
x

FIGURE 1.3 Two surfaces with similar height parameters.

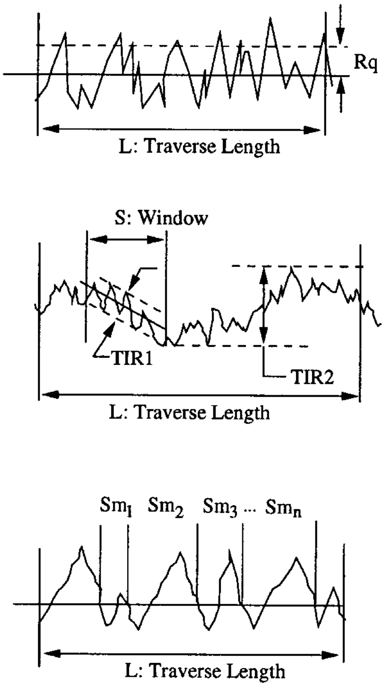


FIGURE 1.4 A subset of surface parameters.

Among the amplitude parameters, the RMS height average, R_q, shown in Figs. 1.3 and 1.4, is the most reliable [74]. It is also the most commonly used on the production floor. However, monitoring an average single value such as the surface RMS roughness can often lead to false or incomplete conclusions about the surface precision [43, 74]. For example, two profiles with different frequency fault patterns with similar amplitude values will result in similar average height values even though the fault mechanisms are completely different, as shown in Fig. 1.3.

Additional profile measures are commonly used to complement the limited information provided by the roughness measures. For example, the waviness parameter W_q is equivalent to R_q: it provides information on the average height characteristics of the filtered waviness profiles [44, 74]. In addition, overall height parameters such as TIR_1, and TIR_2 (total indicator reading), shown in Fig. 1.4, provide the max-minus-min value of the profile. These two parameters are computed based on the least squares best-fit line constructed from the data points of the sample waveform. The purpose is to permit a measurement of straightness of a surface over the total length (TIR_2) or over a short length (TIR_1). Finally, the mean spacing parameter, S_m, shown in Fig. 1.4, is one of the parameters used to describe the mean spacing between profile irregularities at the mean line, distinguishing between profiles with similar amplitude parameters, but different wavelengths.

To complement these average parameters, additional measures are often used to quantify the symmetry and randomness of profiles [14, 43, 44, 74]. The skewness parameter, R_{sk}, is a measure of the symmetry of the roughness profile about the mean line; the measure distinguishes between asymmetrical profiles with the same RMS height values. The kurtosis parameter, R_{ku}, is a measure of the randomness of the profile's shape relative to a perfectly random profile. Such a profile has a kurtosis value of three, whereas a profile with a kurtosis value of greater than three implies less randomness or more repetitiveness.

The examples above represent a subset of additional parameters used to supplement the roughness information. Unfortunately, the addition of parameters does not necessarily provide more reliable information. Using a multitude of parameters often leads to confusion in deciding upon parameters from which conclusions must be drawn [74]. Although an increase in the average profile parameters will indicate the occurrence of a fault pattern, this information is typically not enough to determine the severity of the fault pattern, the specific location, or the character of the fault pattern. An average profile measure will indicate the steady-state values of the faults on the part surface, and not the onset of the problem. This information is crucial and necessary to detect potential problems. and failures in the system. Additional knowledge of the character, or shape of the fault pattern, is often necessary to determine the source of the fault causing a change in surface precision.

Merging Random Process Analysis with Surface Analysis

The use of the parameters above is solidly established in research and in industry. However, they are problematic in analyzing complex signals measured from part surfaces. An average surface parameter, while providing a good first-look at the profile characteristics, is inherently problematic because of its averaging characteristics [67]. Most profiles from manufacturing parts contain a fingerprint of the manufacturing process, and hence can have complex characteristics, such as localized phenomena, freak marks, time-varying trends, etc. Average parameters not only fail to detect these changes in the process, but also fail to indicate the location of such errors. As a result, a more coherent classification based on random process analysis of the surfaces has emerged in metrology research [74]. Complex signals are best analyzed and processed using signal processing tools; these tools are essential in ensuring the correct and efficient analysis of signals from manufacturing systems, such as manufactured surface profiles [49, 75].

Signal Characteristics

In studying the characteristics of signals, we have to first be aware of the different types of signals one can encounter. Signals can be categorized as either deterministic or stochastic. An example of a deterministic signal is a periodic signal [7, 10]. Signals of this nature may be detected easily, since they follow a known model by which exact future values can be predicted. Stochastic signals, on the other hand, do not follow a known model, and their structure and future values are usually described by probabilistic statements. Furthermore, stochastic signals can exhibit stationary and/or nonstationary characteristics. The probabilistic laws describing stationary random signals are time-invariant, i.e., the statistical properties of the signal do not change with time. However, a significant portion of fault-indicating signals from manufacturing machines contain time-varying or nonstationary characteristics, where the statistical parameters change significantly with time, and thus can no longer be predicted by the common techniques [10, 75]. Nonstationary signals can be regarded as deterministic factors operating on otherwise stationary random processes [7, 9]. Nonstationarities can be categorized in the form of three major types [7, 9]: (1) patterns with a time-varying mean value; (2) patterns with a time-varying mean-square value (i.e., variance); and, (3) patterns with a time-varying frequency structure.

Spectral-Based Characterization Methods

One of the major breakthroughs in the characterization of surfaces has been the use of mathematical tools from signal processing applications, specifically the autocorrelation function and the power spectral density function, based on the Fourier transform [74]. The idea of correlation is well known in statistics. The autocorrelation function indicates the degree of dependence of the profile at a given location, on the other profile values displaced by a constant lag [67]. The power spectrum is the frequency domain counterpart of the autocorrelation function [67].

The autocorrelation function and power spectrum have been widely applied to qualitatively distinguish among different profiles and to monitor signals in manufacturin‘g processes [25, 43, 72, 73]. A large number of the applications of the spectral-based methods can be found in the fault monitoring and detection literature. The field of fault monitoring and detection concentrates on finding suitable signal processing methods to extract relevant fault-indicating features in manufacturing, such as the errors on manufactured surfaces. The traditional method of feature extraction relies on the use of the Fourier transform (FT) to decompose a given signal into its different frequency components [8, 20, 28, 37, 41, 64, 74]. The power spectrum analysis based on the Fourier transform is well understood, and is a commonly available tool in commercial signal analyzers. However, the Fourier spectrum method has many limitations that result in an inaccurate picture of the signal decomposition [56, 75], especially when nonstationary features are present. Any difference in the nature of the data function is smoothed out because the total signal is encompassed under the integral sign, creating an averaging effect [75].

Since the power spectrum can fail to provide an accurate picture of the correct frequency decomposition, other methods are commonly used in conjunction with the power spectrum, namely time-domain representations from common random process analysis methods, such as correlation functions, time-domain averages, etc. [37]. A time-domain version of the power spectrum, namely the cepstrum, which is the spectrum of the logarithm of the spectrum of the signal [37, 64], is also sometimes proposed as an aid to Fourier transform. Its main characteristic is that any periodically occurring peaks in the spectrum will appear as a single peak in the cepstrum, thus easily identifiable [53]. Finally, the coherence function, a dimensionless version of the power spectrum, is also used by some researchers [15, 79]. However, these methods do not always help in providing a more accurate picture of the fault condition [74]. Furthermore, they add to the large number of tools proposed for use on the production floor, which makes their acceptance in practice more difficult.

Advanced Mathematical Transforms as Alternatives to the Fourier Transform

The theory of random process analysis is now well established in the field of surface characterization, but it has yet to make an impact in industry [74]. As discussed above, this is due to several factors: the difficulty in interpreting the results; the tendency to obtain misleading or incomplete results due to time-varying or localized surface effects; and the inability to identify when and where exactly the errors take place.

To overcome the above-mentioned shortcomings of the Fourier transform, especially in representing time-varying signals, methods that parameterize time, in addition to frequency, are researched as alternatives to the common random process analysis methods. In general, the aim of signal analysis is to extract relevant information from a signal by transforming it [48]. The Fourier transform is such a tool, resulting in a decomposition into different frequency components based on sines and cosines as basis vectors [8, 28, 74]. However, there exist many other transforms that researchers have explored with the purpose of providing a clearer picture of the signal signature.

The most popular alternative of the Fourier transform is a windowed version of the power spectrum, the short-time Fourier transform (STFT) [74, 77]. The STFT gives a picture of the frequency decomposition of the signal in different time windows, and hence presumably can analyze time-varying signals. However, stationarity still has to be assumed within the short time window, which causes a problem with the frequency/time resolution tradeoff [75].

Other attractive alternatives to the Fourier transform are the Wigner-Ville transform, the wavelet transform, and the higher-order spectral transform. The Wigner-Ville transform has been previously used in analyzing manufacturing signals, including surface signals [46, 49, 75]. However, this transformation typically introduces redundant features when dealing with multi-component signals. This redundancy tends to obscure the significant fault features. The wavelet transform has been used in a number of signal processing applications, including a few manufacturing applications, but there are no applications to surface profiles [22, 30]. Wavelet transforms provide an exciting tool which eliminates the resolution and accuracy problems of other time-frequency decompositions, while providing a more flexible time-scale resolution [22, 30]. However, in the standard formulation of wavelet transforms, it is difficult to attribute physical meaning to the features (wavelet coefficients) extracted from a wavelet decomposition. In our work, we aim to allow physical interpretation of the extracted features to enable understanding of the shape of the fault patterns, their status, and severity. Higher-order spectral transforms have been used in numerous signal processing applications, including several efforts in condition monitoring in manufacturing, with no applications to surface profiles [5, 16]. This method is used to detect unusual frequency components that would not have been detected with the second-order Fourier spectrum, maintaining phase information in addition to the magnitude information provided by the second-order spectral methods. However, it is often difficult to isolate and interpret the features corresponding to nonstationary signals. In addition to these transforms, there exist other transforms, including Gabor transforms, Walsh transforms, and Haar transforms [29, 74], but these are simply extensions of the Fourier, Wigner-Ville, and wavelet transforms, so their utility is similar.

Criteria for Accurate Signal Analysis and Synthesis

To embark on the difficult task of developing improved surface analysis and synthesis methods, we first need to determine a set of criteria for accurate analysis. One of the most crucial tasks in fault detection and monitoring is the proper decomposition of a complex signal into its decorrelated patterns. To obtain a decorrelated decomposition of the measured signal, we need to apply a mathematical transformation to the signal [1, 2, 74]. The mathematical transformation should result in a clear decomposition, which in turn should result in easily-interpretable features. Furthermore, the transformation should prevent any “masking” of the real features by effects such as randomness and nonstationarities. Masking of the actual features may result in the misinterpretation of surface information because of the incompleteness of information, or because of redundancy of information resulting simply from an artifact of the mathematics. As a result, we want to enable the accurate decomposition of all types of patterns, whether deterministic, stochastic, stationary, or nonstationary, into features which are physically meaningful and easily interpretable in the transform domain.

Error prediction requires the establishment of another set of criteria. One of the most crucial tasks in fault prediction is the accurate reconstruction or synthesis of the surface profiles, given the information from the surface analysis method. We can view this problem as having *a priori* knowledge about the manufacturing process and machine characteristics, and knowledge about the part’s functionality, and combining this information to generate a model of the expected surface profile and conditions. To develop a unified method, it is important that a single method be used to perform both tasks of analysis and synthesis simultaneously. To accomplish both tasks, we require that the method be able to take the information from the analysis phase, and reconstruct an estimate of the original surfaces based on this information. As a result, it is crucial that the important information is not lost or masked after the analysis phase, since this can result in the synthesis of inaccurate surface profiles. Finally, it is important that the method provides the mathematical ability to compute an estimate of the original surface profiles. To assure that our methods are easily accepted in practice, we want to avoid the introduction of additional methods to perform the task of synthesis.

Avenues for Improvement

The current status of advanced manufacturing technology requires the automated and accurate detection of any type of errors on part surfaces. Loss of information and/or misrepresentation of the fault condition due to the nonstationary and stochastic nature of surface profiles are not acceptable if we are to engage in automated, fault-free production. In addition, the inability to reconstruct surface profiles based on the information from the analysis phase is also unacceptable, if prediction is to become an essential part of the design-manufacturing cycle. To overcome these common shortcomings, we have developed two novel techniques to decompose and reconstruct manufactured surface profiles [57, 59, 58, 67, 68, 69, 70].

The first technique uses a subset of the traditional surface analysis techniques to remove the deterministic components from the surface profiles. It then analyzes the remaining stochastic profile by means of a multi-resolution wavelet decomposition and quantification with fractal measures. This method of analysis departs from common usage of wavelet transforms by introducing a fractal interpretation. It also provides a complementary technique for surface synthesis, enabling the integration of manufacturing information earlier in the design process. The second technique uses an orthogonal decomposition via the Karhunen-Loève transform to decompose surface profiles into their significant components, and provides a means to monitor individual patterns on surface profiles. The fault status of individual patterns is indicated with this method, with an equal capability to handle various different fault characteristics. The information obtained from the second technique provides a clear understanding of the fault mechanisms; the understanding gained will help designers identify the source of the faults and remedy the problems by redesign and/or process control. The technique allows a means of reconstructing an estimate of surface profiles prior to manufacturing to enable prediction during the design stage.

There are clear distinctions between the two techniques. The advantage of the fractal technique is in its ability to distinguish between stationary or nonstationary stochastic profiles by means of a minimum set of measures. The removal of deterministic trends precedes the fractal analysis of the surfaces. Measures are used to indicate the nature and severity of the deterministic components. However, changes in deterministic components cannot be detected and monitored with this method. The Karhunen-Loève technique has the advantage of decomposing the surface profiles into all of its significant patterns, and monitoring all of the patterns simultaneously, including deterministic and nonstationary patterns, even in the presence of highly stochastic noise. However, the method currently does not allow the monitoring of the stochastic component of a signal. As we discuss below, the conjunction of these two methods is one of the avenues being pursued by the authors.

In the following sections, the details of each method are presented in the context of analysis for surface fault monitoring and synthesis for surface fault prediction. Specifically, the philosophy, models, and the mathematical foundations of each method are presented, followed by applications to numerical signals, as well as to design examples using experimental surface profiles.

1.3 Fractal-Wavelet Method

The first novel technique [57, 59, 58, 67] incorporates traditional spectral-based surface analysis techniques [7] with a fractal-wavelet analysis of stochastic surfaces [76] in the form of a superposition model. This formulation sets the stage for the first component of our vision: characterizing structure in the stochastic nature of manufactured surfaces.

Manufactured surfaces are composed of a kaleidoscopic amalgam of error structures. While the autocorrelation and power spectrum approaches are effective in diagnosing the presence or absence of certain trends, they must be supplemented by additional methods like regression analysis to extract structural parameters, and to synthesize surfaces. One effective tool that is well tailored to deal with the complexity of manufactured surfaces is the fractal dimension. To accomplish our objectives of analysis and synthesis of manufactured surfaces, we use fractal dimensions in conjunction with wavelet transforms. We now survey the background literature on these two tools. Then we present our geometric interpretation of fractal dimensions, followed by a discussion of the pertinence and origin of fractal structures in manufacturing with specific reference to metal cutting processes.

Literature Background: Fractals

The subject of fractals and fractal dimensions was engendered by Mandelbrot in the classical reference to the subject [36]. Several natural objects exhibit fractal characteristics, e.g., coastlines, clouds, and geological formations [51]. There has been an unprecedented surge of applications for fractals in widely diverse fields, ranging from economic data analysis [36], through analysis of lightning patterns [66], to structural studies of human lungs [36].

The study of engineering surfaces has also been influenced by fractals. Gagnepain and Roques-Carmes [21] present a “first approach” to define the surface roughness of a metallized plastic film, in terms of a fractal dimension. Ling [31, 32] has studied the potential applications of fractal geometry in tribology problems. Surface models for boundary lubrication and bio-mechanical joints are used as case studies to illustrate the shortcomings of conventional (deterministic) methods, and possible improvements through the use of fractal geometry.

Majumdar and Bhushan [33] have conducted similar studies on the use of fractal geometry in contact mechanics. The Weierstrass-Mandelbrot (W-M) function [42] is used to characterize the height variations in a surface. A contact theory for fractal surfaces is developed. The theory is used to predict the real contact area between a magnetic tape and flat hard plane, and agreement with experimental results is reported.

The directions taken by the above researchers are all different, with the common theme of using the fractal dimension as a measure of the “error.” Most of them concentrate on the analysis, where the fractal dimension and other parameters are estimated, but few address the inverse problem of synthesizing the profile, with given fractal parameters. Synthesis is a crucial step, as it would enable the designer to assess performance prior to manufacturing. The tool we use for this is the wavelet transform.

Literature Background: Wavelets

The theory of wavelets had its formal genesis from the tools developed for seismic studies [61]. However the scientific antecedents of wavelets have a long history which is well presented by Resnikoff [47]. The basic theory and generalized versions of wavelets are described in [11, 13]. In this research, the theory is restricted to orthonormal wavelets, in view of their analytical simplicity and numerical implementation properties. In this context, we use the formalism of multiresolution analysis [35]. From the surface error viewpoint, this formalism can be interpreted as measuring the form error using a range of diameters on the stylus of a profiling instrument, or studying the different scales of errors through the range of applicable wavelengths [57].

Geometric Notion of Fractal Dimensions

Traditional geometric classification of objects invokes Euclidean dimensions; i.e., a line is one dimensional, a plane is two dimensional and so on. Let us consider a linear element of a flat manufactured surface. Ideally, this surface is a one dimensional profile, which can be represented by a straight line. However, the real profile has crests and ridges (as revealed by a profilometer trace over the element) thus deviating from the ideal profile. This situation is depicted in Fig. 1.5. Focusing on the ideal straight line, we observe that as the errors increase, the structure of the profile also changes. For instance, it fills more space (on the paper) as evidenced by the blackening of the page. In the limit, an extremely irregular profile can fill an area, thus having an Euclidean dimension of 2. This sets the stage for the fractal dimension, D_f, computed as an intermediate dimension (between 1 and 2), used to describe any irregular profile of Euclidean dimension 1. The fractional part of the dimension is an index of the error structure or deviation from the ideal profile. This concept can be extended to describe fractal dimensions between 2 and 3 to characterize the irregularities in planar surfaces of Euclidean dimension 2.

One of the reasons for the wide-ranging applications of fractals is that they capture the essence of the interplay between the geometry and the dynamics underlying the problem [60]. This central idea is our motivation to use fractals in this research, i.e., to study the dynamics of manufacturing processes, and to relate it to the structure of the resultant errors, abstracting it in terms of fractal-based parameters. We now take a closer look at the genesis of fractal structures in manufacturing processes.

Philosophy: Why Fractals in Manufacturing?

Any manufactured surface is the end result of the dynamics of the manufacturing process. Let us consider metal cutting as an example. Several deterministic and random factors interact to produce the surface and the errors manifested therein. Examples of deterministic effects are structural vibration of the machine tool and geometric errors in fixturing; some of the common random effects are hardness

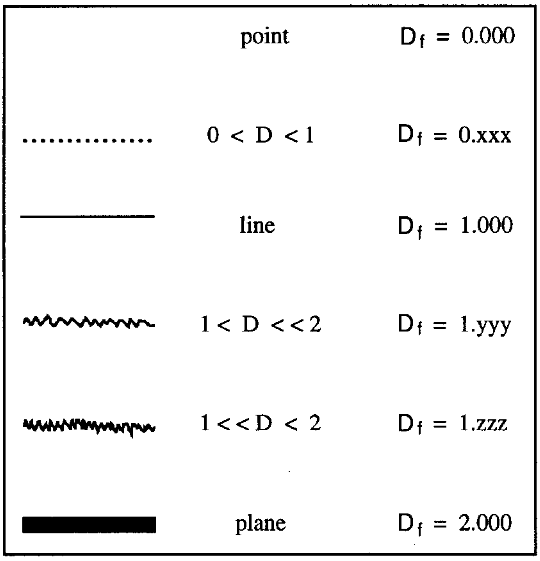


FIGURE 1.5 Irregular geometric objects and their dimensions.

variations in the work material and random vibrations [57]. The intricate interplay of these multifarious dynamic effects lead to distinctive error structures in surface generation.

If we conceive the phase space of the manufacturing process, the dynamics of surface generation can be visualized as a straight line. As the manufacturing process removes (or adds) material, the line evolves. Deterministic effects are convergent and have a propensity to fold the line. Random factors are divergent and are inclined to stretch the same line. From this dynamical synergy, a unique error structure emerges. We conjecture that this error structure is well characterized by fractal parameters. This conjecture is reinforced by shifting from the abstract to the physical, and probing certain specific aspects of machining processes that have the potential to generate fractal profiles/surfaces.

A Laplacian Fracture Model

The primary surface generation mechanism in machining is fracture, caused by nucleation and propagation of cracks. Visualize the workpiece material (prior to any machining) as a matrix of interlinked cells (in the xy plane), with the corners of each cell represented by nodes. As the cutting tool traverses the work material, it displaces the cells in its vicinity, subjecting them to plastic deformation, and ultimately leading to crack formation and chip separation [12]. Therefore each node is a prospective separation point, experiencing a displacement $u\ (x, y)$.

The force equilibrium of any given node is governed by Laplace's equation, (1.1), with suitable boundary conditions [57, 62]:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \tag{1.1}$$

Solving this equation gives the displacement, u. In addition to the displacement caused by the cutting tool, the links between successive nodes also have a fracture probability, depending on local factors, e.g., workpiece hardness. The final surface is a result of these combined effects, and will be fractal [36, 42].

Energy Cascade in Fracture

Retaining the definition of fracture as a crack propagation process, consider the energy associated with the formation of a new surface. In metal cutting, the energy stored in the tool is expended for plastic deformation of the chip and crack propagation, i.e., fracture. In conventional machining analysis, the energy utilized for fracture is ignored [65], as it forms an insignificant proportion of the energy required for plastic deformation. However, a recent investigation on brittle fracture by Gol'dshtein and Moslov [23] highlights the role played by fracture energy in the formation of surface structures.

The crucial interpretation involves considering the workpiece as a hierarchy of scale structures. Griffith's law characterizes ideal brittle fracture as follows: the growth of a crack results in a reduction in the system potential energy, and this is balanced by the work of separation that forms the new surface [18, 40]. However, in the presence of a scale hierarchy the fracture energy forms a cascade, released from larger scales to smaller ones, and finally to microscopic scales, where the new crack is formed. At each of the smaller scales, the energy is modulated as it is passed from larger scales [50]. This energy cascade is described using the following power law [23].

$$G(l_n) \propto l_n^{D_{f-1}}, \tag{1.2}$$

where $G(l_n)$ is the elastic energy released on the nth scale, l_n, and D_f is the fractal dimension.

Premature Deformation in the Tool Path

In metal cutting, high stresses are generated at the tool tip, leading to fracture of the work material. In addition, secondary compressive stresses are generated in the material just ahead of the tool tip. The extent of material affected will depend on the cutting force and the area of stress distribution [26]. Depending on the magnitude of the compressive stress, the material in the tool path will be deformed, elastically or plastically. The portions deforming elastically recover, but those deforming plastically do not, i.e., there is differential recovery along the tool path. Consequently, there exists a varying deformation pattern in the workpiece ahead of the cutting zone. This deformation can conceivably affect the cutting characteristics (e.g., depth of cut), when the relative motion brings the tool to those points. This is yet another possible cause for the existence of interrelated features in surface profiles, which can be characterized by suitable fractal parameters.

Mathematical Implications: Fractal Models

The above subsections have explored the various possibilities for the manifestations of fractal structures in surface errors, characterized by special long range correlations between various scales. An elegant method of modeling the long-term dependence in data is by using fractal parameters [36]. The vehicle we use for computation of these fractal parameters is the theory of wavelets, elucidated in the next section. This also forms the mathematical prelude to the fractal-wavelet relationship, which forms the crux of the analysis and synthesis using this technique.

Fractal-Wavelet Analysis

The objective of the fractal-wavelet analysis procedure is to take any experimental profile and abstract the error information in terms of fractal parameters. This problem is ideally set up within the context of multiresolution analysis. Here the experimental profile is studied at different resolutions, and at each distinct resolution we obtain the approximation and detail. These operations are mathematically defined below.

Approximations

Let us denote a given product surface profile as a function $p(x)$, where x is the measurement coordinate. In manufactured profiles, since the data is usually given in terms of height (from a specified datum) at discrete points along the workpiece, this data sequence is indicated by $A_0^k(p)$. Here A_0 implies the base approximation of the profile p, and k is an integer denoting the kth data point. In general the data sequence corresponding to an approximation at resolution level m is given by $A_m^k(p)$ [67]. Consider a discrete filter H with the following impulse response:

$$h_k = (\phi_{-1}(x), \phi(x-k)), \qquad k \in Z \tag{1.3}$$

where $\phi(x)$ is the scaling function [35]. The mirror filter [35] $\tilde{H}$ is defined as having the impulse response $\tilde{h}_k = h_{-k}$. The approximation at the next coarser level is defined by:

$$A_m^k(p) = \sum_{n=-\infty}^{\infty} \tilde{h}_{2k-n}\, A_{m+1}^n(p). \tag{1.4}$$

This operation is repeated to obtain successively coarser approximations $(-1, -2, -3, \ldots)$, until we are left with one data point, representing the coarsest approximation of the profile.

Details

As we progress from finer to coarser approximations as described above, we lose information about the profile. This vestigial information is captured at each resolution level by the detail. The detail extraction operation is defined as below.

Let g_k be the impulse response of the discrete filter G:

$$g_k = (\psi_{-1}(x), \phi(x-k)), \qquad k \in Z, \tag{1.5}$$

where $\psi(x)$ is the wavelet function [35]. Defining a corresponding mirror filter $\tilde{G}$ with impulse response $\tilde{g}_k = g_{-k}$, the detail extraction operation is written as:

$$D_m^k(p) = \sum_{n=-\infty}^{\infty} \tilde{g}_{2k-n}\, A_{m+1}^n(p). \tag{1.6}$$

Fractal Parameter Computation

Using the above operations, given profile data can be decomposed into approximations and details at various resolution levels. The fractal parameters are computed from the power law relationship governing fractal profiles [57, 76]:

$$\sigma^2[D_m^k(p)] = V_0 2^{-\beta(D_f)m}, \tag{1.7}$$

where $\sigma^2[D_m^k(p)]$ is the variance of the details at resolution m, V_0 is the magnitude factor, and $\beta(D_f)$ is the spectral exponent, which is a function of the fractal dimension, D_f. A regression analysis on the logarithmic version of the above equation yields β as the slope, and $\log(V_0)$ as the y-intercept of the fitted straight line. Depending on the value of β, one of the following two models is used to calculate the fractal dimension [57]:

- Fractional Gaussian Noise (fGn):

$$D_f = \frac{3-\beta}{2}, -1 < \beta < 1. \tag{1.8}$$

- Fractional Brownian Motion (fBm)

$$D_f = \frac{5-\beta}{2}, 1 < \beta < 3. \tag{1.9}$$

The following sections briefly describe the inverse of the decomposition process outlined above.

Fractal-Wavelet Synthesis

Starting with an experimental profile, after decomposition, we have M sets of approximations and details, where M is the number of resolutions possible. We can reconstruct the original profiles using the following procedure:

$$A_{m+1}^{k}(p) = 2\sum_{n=-\infty}^{\infty} h_{k-2n}\, A_{m}^{n}(p) + 2\sum_{n=-\infty}^{\infty} g_{k-2n}\, \mathrm{D}_{m}^{n}(p). \tag{1.10}$$

Thus, starting with the approximation and detail at the coarsest resolution, increasingly finer approximations can be computed until we reconstruct the profile at the base resolution. However, this procedure still requires that we retain all the decomposed data, and is not an efficient synthesis tool. Ideally, we wish to abstract the profile information in a minimum parameter set, which we can use for the synthesis. In this quest, we again resort to Eq. 1.7, and exploit the fractal parameters computed in the analysis. We use a key theorem stated below [76]: reconstruction with discrete signals, which are mutually uncorrelated zero-mean random variables with variances following the power-law yields nearly fractal profiles.

Hence in the synthesis, we generate the details as uncorrelated random variables with variances determined by V_0, $\beta(D_f)$, and m, in accordance with Eq. (1.7). These details are then used along with the coarsest approximation, to construct the hierarchy of finer approximations.

When the above synthesis is applied to actual experimental profiles, certain difficulties are encountered. These difficulties are traced to the presence of strong independent, deterministic effects that influence the computation of the fractal parameters. Consequently we use a novel superposition approach, that combines deterministic and fractal components, while also taking advantage of the diagnostic faculties of traditional surface analysis methods.

A Superposition Approach

Profiles in manufactured surfaces are complex geometrical imprints of a multitude of physical mechanisms involved in the process. Analysis and synthesis of the variegated structural information present in the errors requires a well-stocked mathematical repertoire. In this section, we present a superposition model that invokes, in varying degrees, the autocorrelation, power spectrum, linear and nonlinear regression, and the fractal-wavelet method described above, to achieve our objective.

The first step is to recognize that manufacturing errors possess both deterministic and irregular components. The superposition model proposes an additive combination of these two main components. Further, the deterministic component is broken down into linear and periodic trends which are also added. A tacit assumption in using this model is that the different components are independent of each other. This assumption is reinforced by our experimental observations [57], and previous publications [73]. Another important step, and a key contribution of this research, is using the fractal-wavelet method to characterize the irregular component. Henceforth, the irregular component is called the fractal component.

For purposes of further discussion, the following practical assumption is made. The profile is considered to lie along a horizontal coordinate x, and the error is expressed as $p(x)$. Based on the aforementioned observations, the superposition model is written as:

$$p(x) = p_t(x) + p_p(x) + p_f(x), \tag{1.11}$$

where $p_t(x)$, $p_p(x)$, and $p_f(x)$ are the trend, periodic, and fractal components respectively. While Eq. (1.11) would capture the predominant structural information in the profile, it is possible to obtain a few points that do not conform to this overall structure. In statistical parlance, such unrepresentative points are called "outliers" [6], and for the sake of completeness, they are also added to Eq. (1.11) to obtain:

$$p(x) = p_t(x) + p_p(x) + p_f(x) + p_o(x), \tag{1.12}$$

where $p_o(x)$ is called the outlier component. The next step is to identify each component, define it, and establish methods to compute the relevant parameters for analysis and synthesis.

The linear trend is identified using autocorrelation and power spectrum methods [57, 67]. It is manifested as a slow decay in the autocorrelation, and as a peak at zero frequency in the power spectrum. A linear regression technique is used to compress the trend information in two parameters, I.e., an intercept p_{to}, and a slope s_t.

$$p_t(x) \;=\; p_{t0} + s_t \cdot x \qquad 0 \leq x \leq L. \tag{1.13}$$

A periodic trend exhibits a periodic autocorrelation, and appears as a peak at the underlying frequency in the power spectrum. The periodic component is estimated from the surface profile by using a nonlinear regression procedure. The following model is proposed for evaluating the periodic component $p_p\,(x)$:

$$P_p(x) \;=\; p_{p0} + d_a \cdot \sin(2\pi f_r\, x/L) \qquad 0 \leq x \leq L \tag{1.14}$$

where p_p0 is an offset (from an arbitrary datum of zero), d_a, is the amplitude, and f_r is the frequency. Nonlinear regression methods are used to evaluate the parameters [27, 52]. Initial estimates for p_p0 and d_a are set equal to 0, and the frequency corresponding to the spectral peak is used as a preliminary estimate for f_r.

The fractal component is the remnant from the experimental profile, after subtracting the linear and periodic trends. The outlier data points are corrected using the following rule (57]:

If the absolute value of a data point deviates from the mean by more than twice the standard deviation of the entire data set, then the value is replaced by that of its predecessor. As explained above, the fractal-wavelet method is used to characterize the fractal component in terms of its fractal dimension and magnitude factor. The analysis (extraction of relevant profile parameters from experimental data) and synthesis (regeneration of profiles from the computed profile parameters) is demonstrated for two different processes: milling and grinding.

Milling Profiles

The first process selected for study is a face milling operation, with a flycutter [57]. The flycutter is a single-point tool used to machine light cuts on flat workpieces. Several sources of error are possible in this process: errors from roughing, clamping the workpiece, hardness variations, tool wear, vibrations, and so on.

Experimental Layout

A Bridgeport vertical milling machine is selected. The cutting tool is high speed steel. Work material is Aluminum 2024, with no cutting fluid used. Three process parameters are chosen for this experiment: the cutting speed *s*, the feed *f*, and the depth of cut *d*. Milling is performed at two levels for each factor, leading to 2^3 factorial experiment. The values for the selected factors and levels are shown in Table 1.1. The experiment is replicated twice. The layout of the design of experiments is shown in Table 1.2. Note that the replicate number is indicated by Roman numerals I and II, while the test number is indicated by the letters A through H. In most of the following tables a particular replicate and test number combination is denoted "REP.No", e.g., I.A, II.D.

TABLE 1.1 Factors and Levels for Milling Experiments

Level/Factor	s(fpm)	f(in./rev.)	d(in.)
+	300	0.025	0.02
−	100	0.010	0.01

TABLE 1.2 Layout for Milling Factorial Experiments (2^3)

	Random Order		Factors		
Test No.	I	II	*s*	*f*	*d*
A	1	3	−	−	−
B	8	2	−	−	+
C	5	4	−	+	−
D	7	8	−	+	+
E	3	6	+	−	−
F	4	5	+	−	+
G	6	7	+	+	−
H	2	1	+	+	+

TABLE 1.3 Trend Parameters for Milling

REP. No	Slope	Intercept (mm)	Correlation Coefficient
I.A	–0.000207	0.039692	–0.092
I.B	0.000129	−0.006262	0.114
I.C	−0.000449	0.020916	−0.149
I.D	−0.000450	0.001037	−0.155
I.E	−0.001021	0.007914	−0.917
I.F	−0.000496	0.002156	−0.559
I.G	0.000280	−0.014255	0.125
I.H	0.000818	0.026482	0.290
II.A	−0.001008	−0.025321	−0.395
II.B	−0.000867	0.010203	−0.355
II.C	0.000195	−0.126071	0.028
II.D	0.000618	−0.013617	0.122
II.E	−0.000077	−0.001693	−0.214
II.F	0.000826	−0.000305	0.866
II.G	−0.000663	0.021323	0.116
II.H	−0.000154	−0.028208	−0.027

Data Acquisition and Analysis

The height variations along a linear element of the surface are recorded using a digital electronic indicator. The number of data points is N = 256, to yield eight levels of resolution in wavelet analysis and synthesis; i.e., where resolution levels are in terms of 2^m, $-8 \leq m \leq -1$. Next, the aforementioned methods and tools are used to calculate the pertinent profile parameters for each of the experimental profiles. Table 1.3 shows the linear trend parameters; the periodic and fractal parameters are shown in Tables 1.4 and 1.5, respectively.

Grinding Profiles

Surface grinding is the second operation studied [57, 67]. The grinding wheel can be considered as an agglomeration of multiple cutting edges, with each abrasive grain acting like a single point tool. However, the geometry of the cutting tools is inconsistent, due to the random orientation of the individual grains. Therefore, the grinding process has a predominant stochastic component. Deterministic effects due to vibration are also present [34]. The effect of premature deformation of the workpiece, ahead of the cutting region, are also cited in the same reference. The combination of these numerous erros mechanisms produces complex structural effects in the error profiles

Experimental Design and Conditions

For the experiments in this study, aluminum is chosen as the work material, to enable inter-process comparisons with milling. The grinding experiments are carried out on a Brown & Sharpe 618 Micromaster

TABLE 1.4 Periodic Parameters for Milling

	Frequency (× N cps)		Amplitude	Offset	Sum of Squares (mm^2)	
REP. No.	Initial	Final	(mm)	(mm)	Initial	final
I.A	50	50.162	−0.014537	0.037047	0.423	0.044
I.B	49	49.136	0.009259	−0.004608	0.024	0.007
I.C	29	29.265	−0.020047	0.015253	0.186	0.075
I.D	29	29.345	0.028065	−0.004871	0.123	0.017
I.E	86	85.832	0.004031	−0.000011	0.003	0.0007
I.F	86	85.603	0.004738	−0.000014	0.008	0.005
I.G	34	33.926	0.020302	−0.010715	0.100	0.018
I.H	35	35.302	−0.021169	0.036974	0.460	0.054
II.A	46	46.118	0.012049	−0.038165	0.464	0.073
II.B	46	46.226	−0.020527	−0.000849	0.084	0.030
II.C	27	27.421	0.057442	−0.124114	4.580	0.247
II.D	28	28.070	−0.039048	−0.005752	0.366	0.162
II.E	86	86.012	−0.002095	−0.002671	0.004	0.001
II.F	86	86.108	0.001242	0.0000008	0.003	0.003
II.G	31	30.590	−0.045072	0.013334	0.500	0.198
II.H	33	33.076	0.047913	−0.030261	1.147	0.151

TABLE 1.5 Fractal Parameters for Milling

REP No.	Spectral Exponent	Fractal Dimension D_f^Δ	Magnitude Factor (mm^2)	Correlation Coefficient
I.A	−0.778524	1.889262	0.000122	−0.948
I.B	−0.576505	1.788253	0.000013	−0.945
I.C	−1.527652	2.263826	0.000336	−0.807
I.D	−0.761165	1.880583	0.000041	−0.987
I.E	−0.804503	1.902252	0.000001	−0.866
I.F	−0.705957	1.852979	0.000011	−0.908
I.G	−0.935501	1.967751	0.000056	−0.900
I.H	−0.458841	1.729421	0.000077	−0.739
II.A	−0.767619	1.883810	0.000107	−0.729
II.B	−0.584258	1.792129	0.000038	−0.798
II.C	−1.458700	2.229350	0.001951	−0.952
II.D	−0.798113	1.899057	0.000460	−0.918
II.E	−0.223818	1.611909	0.000001	−0.567
II.F	−0.088341	1.544171	0.000001	−0.154
II.G	−1.018149	2.009075	0.000767	−0.940
II.H	−1.156775	2.057839	0.000728	−0.950

TABLE 1.6 Factors and Levels for Grinding Experiments

Level/Factor	V (fpm)	v (fpm)
+	6259	53.625
−	3500	6.094

surface grinding machine, using single pass grinding. Based on previous studies of the grinding operation [17], two factors are considered, each at two levels: the peripheral speed of the wheel (V) and the work table (reciprocating) speed (v). This constitutes a 2^2 factorial design [45]. The values selected for the factors are presented in Table 1.6 in appropriate units. The wheel speed is changed by using different sizes of grinding wheels. The grinding wheel specification is A46HV, which implies a medium grit, medium (hardness) grade, vitrified bond, alumina wheel. Depth of cut is fixed at 0.0005 in. No coolant is used. Replication and randomization are conducted similar to milling.

TABLE 1.7 Layout for Grinding Factorial Experiments (2^2)

Test No.	Random Order I	Random Order II	Factors V	Factors v
A	2	3	−	−
B	1	4	−	+
C	4	1	+	−
D	3	2	+	+

TABLE 1.8 Trend Parameters for Grinding

REP No.	Slope	Intercept (μm)	Correlation Coefficient
I.Aa	0.012156	1.752679	0.399
I.Ab	0.019312	1.748268	0.473
I.Ba	0.020890	−0.885780	0.677
I.Bb	0.011962	0.713972	0.493
I.Ca	−0.011039	−0.815722	−0.359
I.Cb	−0.022987	−2.522975	−0.698
I.Da	−0.385428	1.496771	−0.842
I.Db	0.011787	0.588932	0.673
II.Aa	−0.003322	−1.894023	−0.146
II.Ab	0.012707	0.121255	0.600
II.Ba	0.023930	0.278721	0.779
II.Bb	−0.019406	1.086330	−0.700
II.Ca	−0.029827	−0.178365	−0.805
II.Cb	0.006601	0.821609	0.276
II.Da	0.009690	−0.045278	0.570
II.Db	0.002408	1.142413	0.158

The layout of the experimental design is shown in Table 1.7. Again, replicate numbers are indicated by Roman numerals I and II, while test numbers are indicated by the letters *A* through *D*.

Data Acquisition and Analysis

The grinding variations are measured with a profilometer. Number of data points is $N = 256$, as in milling. Two different measurements are made for each test piece (errors measured along two different elements on the surface), the first measurement is denoted '*a*', and the second '*b*'. For example, II. Da implies the first measurement (*a*) on the fourth test piece (*D*) of the second replicate (II).

The aforesaid mathematical techniques are again used to calculate the profile parameters for each grinding profile. The linear trend parameters and the fractal parameters are displayed in Tables 1.8 and 1.9, respectively. Conspicuously, the periodic component is missing, as there is no indication of periodic structure either from the autocorrelation or the power spectrum. This makes intuitive sense, as randomness prevails in ground surfaces [73].

Comments on the Analysis

The characteristics of milling and grinding profiles, as exemplified by the profile parameters, are summarized as below.

- The milling profiles show a significant linear component only in a few cases, possibly due to errors in clamping the workpieces. However, there is a significant trend component in all grinding profiles. In view of this consistency, the possible cause is a static error in the vise, machine table, or foundation in either the pre-machining operation or the grinding operation.

TABLE 1.9 Fractal Parameters for Grinding

REP. No	Spectral Exponent	Fractal Dimension D_f^Δ	Magnitude Factor (μm^2)	Correlation Coefficient
I.Aa	0.456197	1.271902	0.158817	0.697
I.Ab	0.508043	1.245979	0.188157	0.959
I.Ba	0.640279	1.179861	0.103456	0.999
I.Bb	0.425596	1.287216	0.120276	0.983
I.Ca	0.823478	1.088261	0.136429	0.995
I.Cb	0.696449	1.151776	0.110229	0.982
I.Da	0.185799	1.407101	0.062795	0.355
I.Db	0.162700	1.418650	0.065793	0.522
II.Aa	0.464151	1.267925	0.079817	0.794
II.Ab	0.130880	1.143456	0.073059	0.336
II.Ba	0.035566	1.482217	0.239665	0.066
II.Bb	0.102393	1.448804	0.244190	0.309
II.Ca	0.494507	1.252747	0.065129	0.862
II.Cb	0.427942	1.286029	0.067165	0.887
II.Da	0.508013	1.245994	0.030870	0.794
II.Db	0.504878	1.247561	0.057347	0.958

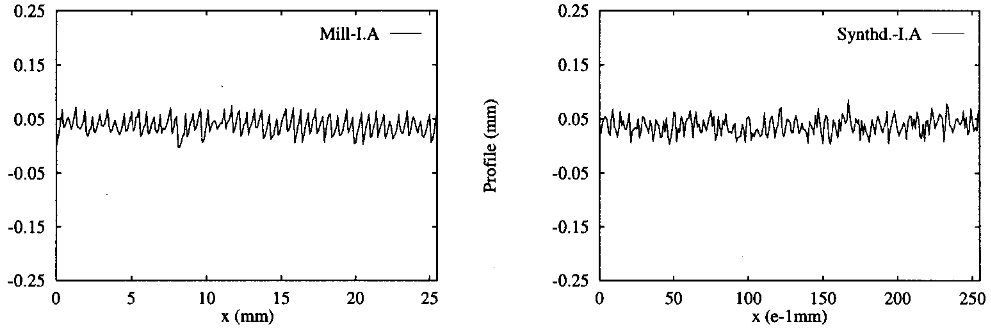


FIGURE 1.6 Experimental and synthesized profiles: Mill-I.A.

- A notable periodic component is observed in all milling profiles; moreover, there is a strong correlation between the frequency of the periodic component and the feed used; e. g., profiles A and B have the same feed, and comparable frequency. There is no periodic component in any of the grinding profiles, in contrast to milling. This is expected, as there is no incremental feed in single-pass grinding as in milling.
- The fractal dimensions are in the range $1.5 < D_f < 2.0$ for milling, and $1 < D_f < 1.5$ for grinding. This indicates that the grinding profiles have relatively fewer (but still important) irregularities compared to the milling profiles. A similar statement can be made based on the magnitude factor as well. This deduction conforms to practical expectations, as grinding is a finishing process, while milling is a roughing/semi-finishing process.

Synthesis and Comparison

The above parameters are used to synthesize the corresponding profiles. Two example experimental profiles, one each from milling and grinding, and the corresponding synthesized versions are shown in Figs. 1.6 and 1.7. It can be seen that the synthesized profiles have similar characteristics, when compared to the experimental. In addition to the visual similarities, the energy and the entropy of the original and synthesized profiles compare favorably as well [57].

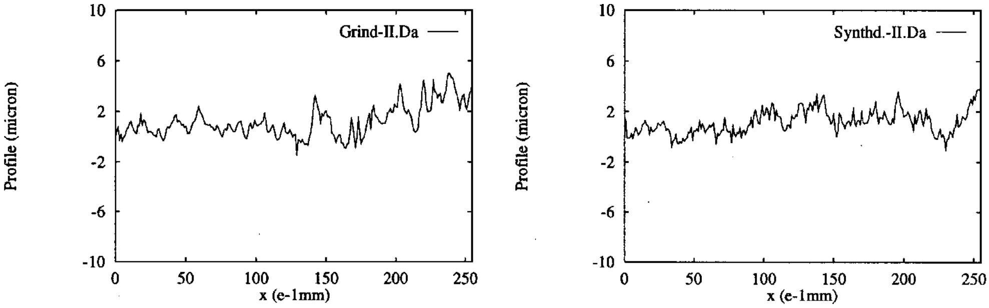


FIGURE 1.7 Experimental and synthesized profiles: Grind-II.Da.

Performance Comparison: Axial Vibrations in Ball Bearings

The goal of this section is to bolster the validity of the fractal-wavelet method with an application to a design example, namely the tolerance design for rolling elements in ball bearings. The salient features of the design problem are introduced below.

Ball bearings are critical machine elements used to transmit power and to support loads. Typical applications are machine tool spindles, power tools, etc. The specific problem under study concerns bearings used to support axial loads, and the influence of form errors in the bearing elements on the axial vibration force. These form errors induce Hertzian contact forces, which augment the axial force. Large axial vibrations can lead to catastrophic failures.

Analysis of Bearing Forces

Figure 1.8 shows the components in an angular contact ball bearing. It is assumed that the applied load to the bearing is purely a thrust load. The performance parameter of interest is the axial vibration force. Additional assumptions made in calculating bearing forces are: the bearing has N_b equispaced balls, firmly in contact with the inner and outer races; the rotational speed of the bearing is moderate, implying negligible centrifugal forces; and the outer race is stationary. Geometric errors in the rolling elements result in both axial and radial vibration forces [71].

In this section, the error is synthesized using the theory developed above and is superposed on the circumference of the outer race. The inner race and the balls are considered to have ideal geometries.

The equations for the axial vibration force can be derived from the Hertzian contact forces. The normal load Q_j on a ball j in terms of the corresponding deflection δ_j is:

$$Q_j = C_f \delta_j^{3/2}, \qquad 1 \le j \le N_b, \tag{1.15}$$

where C_f is the contact compliance factor. The contact deflection δ_j is influenced by the geometric errors in the outer race, and is hence expressed as a nominal deflection δ_0 (i.e., in the absence of any geometric error) and a time-varying component $\delta_j(t)$, which is dependent on the relative position of the ball j, and the outer race:

$$\delta_j = \delta_0 + \delta_j(t) \tag{1.16}$$

Equation (1.15) is approximated assuming a constant C_f [71] as below:

$$Q_m = \kappa \delta_0^{3/2} \left\{ 1 + \frac{3\delta_j(t)}{2\delta_0} + \frac{3\delta_j^2(t)}{8\delta_0^2} \right\} \tag{1.17}$$

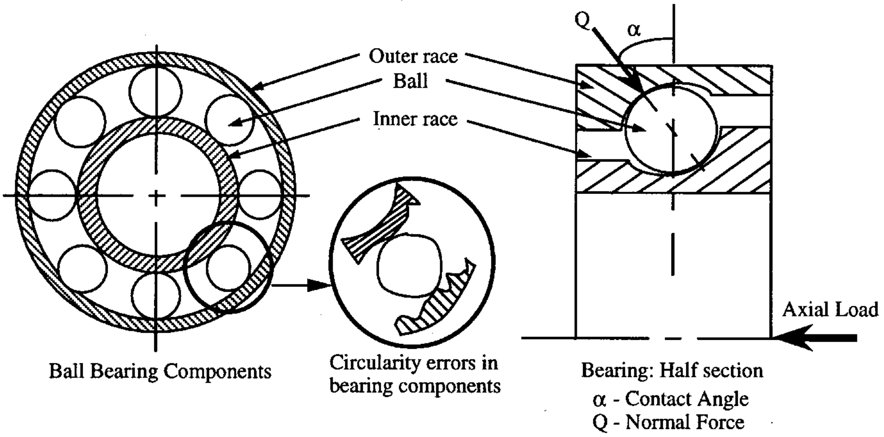


FIGURE 1.8 Angular-contact ball bearing: geometry and forces.

TABLE 1.10 Data for Test Bearing

Parameter	Value
Type	Angular-contact
No. of balls	16
Ball diameter	1.40 mm
Pitch circle diameter	6.75 mm
Radial clearance	0.032 mm
Test load (axial)	1000 N
Test speed	1800 rpm
Contact deflection δ_0	0.0092 mm
Compliance factor C_f	$9.56 \times 10^9 \text{Nm}^{2/3}$
Contact angle α	18.8°

The total axial force on the bearing is the sum of the axial components of the normal forces on all the balls (Fig. 1.8):

$$P_a = \sum_{j=1}^{N_b} Q_j \sin \alpha \tag{1.18}$$

where α is the contact angle. These equations are used in the calculation of axial vibration force for a test bearing. The data for the test bearing are adapted from [71], and are shown in Table 1.10.

Vibration Force with Synthesized Profiles

The experimental and synthesized errors from milling and grinding are superposed on the ideal profile of the ball bearing outer race and the resulting deflection is calculated according to Eq. (1.16), i.e., $\delta_j(t)$ is a contribution of the profile error. Equation 1.18 is used to calculate the axial force at various points in time, from $0 < t < T_0/N_b$, where T_0 is the time for one revolution of the bearing. The force pattern will be periodic with period T_0/N_b, as the balls and the inner race are assumed to have ideal profiles. The results for two profiles, one each from milling and grinding, are shown in Fig. 1.9.

This figure enables the *performance comparison* of experimental and synthesized profiles. The comparisons show good agreement with respect to absolute values and range of variation for both processes considered.

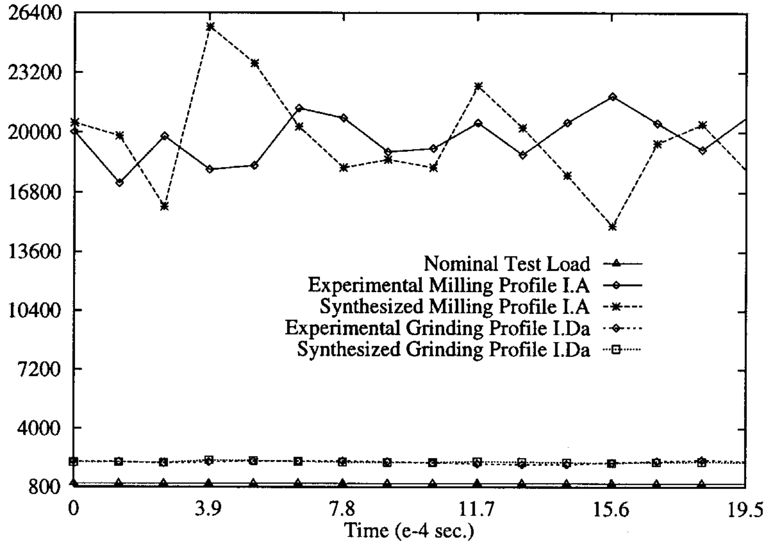


FIGURE 1.9 Axial vibration forces for synthesized and experimental profiles, also depicting grinding and milling processes.

This simple design example also illustrates that the proposed method can be a useful tool for the designer to assess the impact of different manufacturing processes on the performance of mechanical components. For example, it can be seen that milling the outer race will lead to excessive axial vibrations, which in turn can cause premature failure due to very high contact stresses, and that a finishing process like grinding is mandatory to obtain low vibration forces. While this is an obvious conclusion, the same line of reasoning can be extended to more complex problems.

1.4 Karhunen-Loève Method

The second novel technique extends an orthogonal transform, namely, the Karhunen-Loève transform of surface profiles in order to develop a means of monitoring individual patterns and detecting the occurrence of faults on part surfaces [67, 69, 70]. Fault information will be fed back into the design process to help integrate the design and manufacturing cycles. The following sections present the philosophy and details of an analysis and synthesis method based on the Karhunen-Loève transform. This philosophical foundation sets the stage for the second component of our vision: characterizing the deterministic aspects of manufactured surfaces in the presence of stochastic and time-varying effects.

Philosophy

The set of analysis methods discussed in this work all attempt to detect error signatures in signals. Two major approaches can be identified: (1) extraction of previously known patterns in signals, such as linear trends and sinusoidal components, by projecting the data onto pre-determined components; this is the case in Fourier-based methods, for example, where the idea is to expand the data in terms of sines and cosines; (2) detection of error occurrence by overall measures that are sensitive to effects such as non-stationarities; this is the case of fractal measures, for example. What if we approached this problem from a different point of view? Instead of identifying patterns whose shape and characteristics we know, what if we could identify any type of pattern with a single method, no matter what the shape of the pattern is? This would require the use of a transform where the data is not projected onto known functions, such as sines and cosines. In addition, instead of identifying errors by average measures of potentially multi-component signals, what if we could decompose the signal into decorrelated patterns and develop measures for each individual pattern separately? The information we obtain from such measures will not have the risk of containing ambiguous information. With such a tool, not only could we decompose surface signals into their significant components, but also use this tool to monitor changes in each of these components with the purpose of diagnosing fault mechanisms in the manufacturing system.

With these thoughts in mind, we have explored the existence of a mathematical transform that would provide us with these idealistic results. The Karhunen-Loève transform is such a candidate. This transform decomposes the signals into completely decorrelated components in the form of empirical basis functions that contain the majority of the variations in the original data. An estimate of the original signal is computed using a linear combination of these empirical basis functions and their respective coefficients in the new transform domain. The following first presents the fundamentals and results involved with the surface analysis stage. Surface analysis represents the first step into understanding the mechanisms which result in faults during manufacturing. We then present a possible scenario in design using an industrial application. Here, we show that the analysis information from manufacturing can be integrated successfully into the design process. Furthermore, we demonstrate the power of our method in understanding new manufacturing processes. When faced with a new manufacturing process, communication between manufacturers and designers becomes crucial in identifying the existing and potential precision and accuracy problems.

Literature Background

The Karhunen-Loève transform (KL) is used in a variety of signal processing applications [2, 3, 4, 19, 24, 38, 54, 80], mainly as a means of detecting "dominant" characteristics in a set of signals. Here, we present a small subset of the example applications of the transform.

In the field of speech signal analysis and synthesis, Zahorian uses Principal Components Analysis (PCA) to select the most dominant characteristics from the spectral analysis of speech signals [80]. This study exploits the experimentally observed correlations among spectral band energies at different frequencies in order to derive a much smaller set of statistically independent parameters (principal components) which retain most of the information present in the original speech spectra [80]. Sirovich et al. apply the Karhunen-Loève transform to pattern recognition, attacking the general problem of characterizing, identifying, and distinguishing- individual patterns drawn from a well-defined class of patterns [541. The patterns are pictures of faces of people, taken from a random sample of males. They show that a compressed version of the original faces can be reconstructed with reduced dimensionality to generate recognizable faces [54]. Ball et al. use the Karhunen-Loève transform in the analysis of low-Reynolds number turbulent channel flow. Snapshots of the velocity field $u(x,t)$ are taken and the entire ensemble of velocity measurements are decomposed into spatial modes (basis eigenvectors) and time-dependent amplitude coefficients [4]. Graham et al. use principal orthogonal decomposition to decompose spatiotemporal signals into orthogonal spatial components and time-dependent amplitudes. Patterns are obtained during the oxidation of hydrogen on a nickel disk in a mixed flow reactor [24]. Finally, Algazi et al. use the Karhunen-Loève transform to capture the time-varying structure of the spectral envelope of speech. Acoustic subword decomposition using fast Fourier transform (FFT) and the Karhunen-Loève transform are used to extract and efficiently represent the highly correlated structure of the spectral envelope [3].

In manufacturing, the applications of Karhunen-Loève are rare. One example is the use of PCA as a means of multivariate statistical process control [38]. In this work, a multitude of parameters (such as color, texture, strength, temperature, speed, etc.) are measured in a plant and considered simultaneously to detect variations in the parameters. The Karhunen-Loève transform of manufacturing signals has never been proposed as an alternative fault monitoring technique. This is partially due to the difficulty in understanding the physical significance of the resulting outputs in the transform domain. In the following, we first present the mathematical basis of the Karhunen-Loève transform. Then we present our extensions to the transform in an attempt to attribute physical significance to its outputs. These extensions constitute the foundation for developing an effective fault monitoring and detection tool [67, 69, 70].

Mathematical Background

In this subsection, we summarize the background of the Karhunen-Loève approach. A random process $x(t)$, defined in the time domain (or spatial domain) $(0,T)$, can be expressed as a linear combination of orthogonal functions [19]:

$$x(t) = \sum_{i=1}^{N} y_i \varphi_i(t), \tag{1.19}$$

where the orthogonal functions $\varphi_{\mathrm{i}}(t)$ are deterministic functions, and the ith coefficient vector y_i contains random variables y_j. Eq. (1.19) is the Karhunen-Loève expansion of the original data, composed of a linear combination of the basis functions and the feature coefficients in the transformed domain.

To use this approach with real signals in the discrete domain, we take N time-sampled values of the time functions and convert them to vectors as shown in the following equations [19]:

$$X = [x(t_1)\ldots x(t_N)]^T \tag{1.20}$$

$$\Phi = [\varphi_i(t_1)\ldots \varphi_i(t_N)]^T, \tag{1.21}$$

where each time-sampled value $\mathrm{x}(t_i)$ is a random variable.

Assuming zero-mean random processes, the covariance matrix equals the autocorrelation matrix, and is computed as $R(t,\tau) = E[x(t)x^*(\tau)]$. If $\varphi_i(t_k)$ are the eigenfunctions of $R(t_l,t_k)$, they must satisfy the following characteristic equation:

$$\sum_{k=1}^{n} R(t_l, t_k)\varphi_i(t_k) = \lambda_i \varphi_i(t_l), \tag{1.22}$$

where $i, l = 1, 2, \ldots, N$. Equation (1.22) can be written in matrix form to define the eigenvalues λ_i and eigenvectors Φ_i as follows:

$$S\Phi_i = \lambda_i \Phi_i, \tag{1.23}$$

where $i = 1, 2,\ldots, N$ and S is the $N \times N$ covariance matrix defined by:

$$S = \begin{bmatrix} R(t_1, t_1)\ldots R(t_1,t_N) \\ . \qquad\qquad . \\ . \qquad\qquad . \\ . \qquad\qquad . \\ R(t_N,t_1)\ldots R(t_N,t_N) \end{bmatrix} \tag{1.24}$$

The covariance matrix equation is solved to obtain its eigenvalues and corresponding eigenvectors. The eigenvectors are orthonormal and hence need to satisfy $\Phi^T\Phi = I$, where I is the identity matrix. The eigenvalues are ranked in order of significance, as compared to the total energy represented by the sum of all eigenvalues. The eigenvectors corresponding to the dominant eigenvalues represent the fundamental functions in the transformed domain. The Karhunen-Loève coefficients y_i are computed by projecting the original data onto the new domain, represented by the dominant basis functions:

$$y_i = \Phi_i^T X, \tag{1.25}$$

where we have one coefficient vector $y_i = [y_1\ldots y_M]$, corresponding to each eigenvector, and where M is the number of input snapshots. The resulting vector $Y = [y_i\ldots \mathrm{y}_N]^T$ contains the N coefficient vectors corresponding to the N eigenvectors $\Phi = [\varphi_1 \ldots \varphi_N]^T$. Y is an orthonormal transformation of the random vector X into the new vector space, and is itself a random vector. We can reconstruct the original data in the new transformed function domain as a linear combination of the fundamental eigenvectors and corresponding coefficient vectors as in $X = \Phi Y = \Sigma_{i=1}^{m} y_i \varphi_i$, where $m<N$ is the number of fundamental eigenvectors.

Karhunen-Loève Transform for Surface Analysis

In order to understand the mechanics of the Karhunen-Loève method, it is important to understand each output from the method, and determine the manner in which they can be used for our purposes. The intrinsic model used with the Karhunen-Loève transform is a linear combination of orthogonal patterns. The transform results in decorrelated and orthogonal functions that represent the data with the most dominant characteristics. These orthogonal functions are the eigenvectors of the covariance matrix. The covariance matrix contains all the variability in the data. In addition, the eigenvectors depend on the variation of each input vector from the ensemble mean vector. As a result, any deviation from the normal or ideal characteristics of the manufacturing signal will result in additional eigenvectors. The changes in each of these fundamental patterns in the signal are detected by plotting the coefficients corresponding to each eigenvector with respect to the input vectors. If the inputs are taken at regular time intervals, the resulting plot will indicate the change in the fundamental eigenvectors with respect to time. The eigenvalues of the covariance matrix corresponding to each eigenvector will indicate the variance of the coefficients for each eigenvector, as well as the significance of the total energy contained in each eigenvector.

The mathematical development above presents the case for a single vector which is discretized in time (or space) [70]. In our development, we assume that regular snapshots of the process (e.g., manufacturing surface) being monitored are obtained in order to detect possible changes in the process. These changes are assumed to be indicative of fault occurrences.

Given input data $X = [x(t_1) \ldots x(t_N)]^T$, the covariance matrix of X is $S_X = E\,[(X - \bar{X})(X - \bar{X})^T]$ where $E[\cdot]$ is the expected value operator, and $\bar{X}$ is the mean vector $E[X]$. We remove the mean vector from the input vectors, transforming the input data into zero-mean input data, i.e., $\bar{X} = 0$. We have $j = 1 \ldots M$ input data vectors X_j, each of $i = 1 \ldots N$ sampled random variables. These represent the snapshots of the process being monitored, taken at regular time intervals. The input matrix X is composed of these M snapshots X_j. Estimating the expected value operator as an ensemble average, the covariance matrix is then computed as:

$$\begin{aligned}
\hat{S} &= \frac{1}{M}\sum_{j=1}^{M} \mathrm{X}_j \mathrm{X}_j^T \\
&= \frac{1}{M}\left(\begin{bmatrix} x_{11} \\ x_{12} \\ \vdots \\ x_{1N} \end{bmatrix} [x_{11}x_{12}\ldots x_{1N}] + \cdots + \begin{bmatrix} x_{M1} \\ x_{M2} \\ \vdots \\ x_{MN} \end{bmatrix} [x_{M1}x_{M2}\ldots x_{MN}] \right) \\
&= \frac{1}{M}\left(\begin{bmatrix} x_{11}^2 + \cdots + x_{M1}^2 & x_{11}x_{12} + \cdots + x_{M1}x_{M2} & . & . \\ . & x_{12}^2 + \cdots + x_{M2}^2 & . & . \\ . & & . & . \\ . & & . & . \\ . & & . & x_{1n}^2 + \cdots + x_{MN}^2 \end{bmatrix} \right).
\end{aligned} \tag{1.26}$$

The diagonal terms represent the average variances of the individual random variables, while the off-diagonal terms represent the average covariances between two random variables. The first term of the first row in the matrix is the variance of the first point for each input vector, the second term is the covariance between first and second sampled points for each vector, the third term is the covariance between the first and third sampled points for each input vector, etc. Note that if the input vectors are not zero-mean, these terms are interpreted as the mean-squared deviations from the mean vector.

The resulting N by N covariance matrix is real and symmetric, and contains the mean-squared deviations from the mean vector. If the input vectors are zero-mean, then the matrix contains mean-squared deviations between each input vector. As a result, the matrix contains all of the variational structure of the input data. The eigenvectors of this matrix provide the principal axes of the variational structure, presented in the form of basis functions representing the dominant patterns in the data. The eigenvectors are used here to determine the dominant patterns in the monitored signal. The eigenvalues of the covariance matrix are computed using $det(S - \lambda I) = 0$, where I is the identity matrix; λ_i are the roots of the characteristic polynomial, representing the eigenvalues of the covariance matrix. The eigenvectors Φ_i of the matrix are then computed using $(S - \lambda_i I)\ \Phi_i = 0$. We will show that these eigenvectors are dependent on the deviations of each point of the input vectors from the mean vector.

Even though the covariance matrix has dimension N, the rank of the matrix is determined by the number of input vectors M, resulting in M independent eigenvectors of dimension N. Each eigenvector Φ_i has a coefficient vector y_i associated with it, computed using $y_i = \Phi_i^T X$. The coefficient vectors of dimension M are the projection of the original data matrix X onto the corresponding eigenvectors, and hence represent the weight of each input vector in the new transform domain spanned by the eigenvectors Φ_i. As a result, the coefficient vectors are used here to monitor the changes over time for each dominant eigenvector.

Decomposing Complex Signals into Orthogonal Functions

To investigate the ability of the Karhunen-Loève transform to decompose signals, we assume that a general function $g(x,t)$ represents a manufacturing signal, such as a surface profile. This function is composed of a multitude of functions $f_1(x, t)$, $f_2(x, t)$, etc. In reality, the exact shape of the functions f_i, and the exact nature of the interactions between these functions are not known *a priori*. The decomposition of $g(x, t)$ into individual functions f_i will ultimately enable accurate monitoring.

Simple Linear Functions

To demonstrate the mechanics of the Karhunen-Loève method, we first study the case of simple linear vectors as our general function [70]. Linear vectors are the simplest form of deterministic functions. Two sample input vectors X_1, and X_2, each with two sample points $(x_{11}\ x_{12})^{\mathrm{T}}$ and $(x_{21}\ x_{22}\ )^{\mathrm{T}}$, respectively, are collected, as shown in Fig. 1.10 ($M = 2$, $N = 2$). The input vectors represent pure linear trends (straight lines), of increasing slopes. The mean vector $X_a = (x_{a1} x_{a2})^T$ is also a straight line, with two sampled points. δ_{ij} is assumed to be the deviation of the ith sampled point of the jth input vector from the mean vector. In this case, since we only have two input vectors, the average vector is equidistant from each input vector, i.e., $\delta_{11} = \delta_{12}$ and $\delta_{21} = \delta_{22}$, as shown in Fig. 1.10. We first show that, given $M = 2$ linear vectors with $N = 2$ sampled points each, the Karhunen-Loève transform results in a single fundamental eigenvector $\Phi = [\varphi_1\ \varphi_2]^{\mathrm{T}}$, Which is a linear vector.

To demonstrate this result, let $\mathrm{X}_1 = [x_{11} x_{12}]^T$ and $\mathrm{X}_2 = [x_{21} x_{22}]^T$ be two linear vectors. The mean vector is $X_a = (x_{a1} x_{a2})^T = \left[\frac{x_{11} + x_{21}}{2} \frac{x_{12} + x_{22}}{2}\right]^T$. Subtracting the mean vector from the input vectors, we obtain the zero-mean input vectors, used to compute the covariance matrix: $Z_1 = X_1 - X_a = [x_{11} - x_{a1} x_{12} - x_{a2}]^T$ and $Z_2 = X_2 - X_a = [x_{21} - x_{a1} x_{22} - x_{a2}]^T$. Assuming δ_{ij} deviations of the input vectors from the mean vector, the zero-mean input vectors become: $Z_1 = [- \delta_{11} - \delta_{21}]^T$ and $Z_2 = [\delta_{11} \delta_{21}]^T$, which results in the data matrix $\mathrm{Z} = [\mathrm{Z}_1\ \mathrm{Z}_2]$.

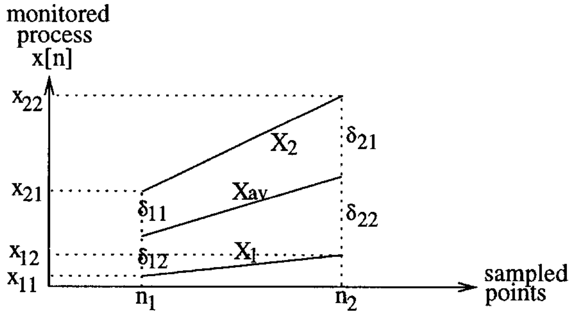


FIGURE 1.10 KL decomposition of linear vectors.

The covariance matrix is computed using: $\hat{S} = \frac{1}{2}\Sigma_{i=1}^{2} Z_i Z_i^T$, which becomes:

$$\hat{S} = \frac{1}{2}\left[\begin{bmatrix}-\delta_{11}\\ -\delta_{21}\end{bmatrix}[-\delta_{11} - \delta_{21}] + \begin{bmatrix}\delta_{11}\\ \delta_{21}\end{bmatrix}[\delta_{11}\delta_{21}]\right] = \begin{bmatrix}\delta_{11}^2 & \delta_{11}\delta_{21}\\ \delta_{21}\delta_{11} & \delta_{21}^2\end{bmatrix}. \qquad (1.27)$$

The eigenvalues are computed using $det(S - \lambda I) = 0$, which becomes:

$$\begin{vmatrix}(\delta_{11}^2 - \lambda) & \delta_{11}\delta_{21}\\ \delta_{21}\delta_{11} & \delta_{21}^2 - \lambda\end{vmatrix} = 0 \quad \Leftrightarrow (\delta_{11}^2 - \lambda)(\delta_{21}^2 - \lambda) - \delta_{11}^2\delta_{21}^2 = 0$$

$$\Leftrightarrow \lambda^2 - (\delta_{11}^2 + \delta_{21}^2)\lambda = 0 \qquad (1.28)$$

The solution of this characteristic polynomial in λ is $\lambda_1 = 0$ or $\lambda_2 = \delta_{11}^2 + \delta_{21}^2$, resulting in only one dominant eigenvalue. The corresponding eigenvector is computed using $(S - \lambda_i I)\Phi_i = 0$, which becomes:

$$\begin{bmatrix}\delta_{11}^2 - (\delta_{11}^2 + \delta_{21}^2) & \delta_{11}\delta_{21}\\ \delta_{21}\delta_{11} & \delta_{21}^2 - (\delta_{11}^2 + \delta_{21}^2)\end{bmatrix}\begin{bmatrix}\varphi_1\\ \varphi_2\end{bmatrix} = 0 \quad \Leftrightarrow \begin{bmatrix}-\delta_{21}^2 & \delta_{11}\delta_{21}\\ \delta_{21}\delta_{11} & -\delta_{11}^2\end{bmatrix}\begin{bmatrix}\varphi_1\\ \varphi_2\end{bmatrix} = 0$$

$$\Leftrightarrow -\delta_{21}^2\varphi_1 + \delta_{11}\delta_{21}\varphi_2 = 0$$

$$\Leftrightarrow \varphi_1 = \frac{\delta_{11}}{\delta_{21}}\varphi_2 \qquad (1.29)$$

which is the equation for a straight line. Since the eigenvector has to be orthonormal, the orthonormality condition $\Phi^T\Phi = \mathbf{I}$ implies $\varphi_1^2 + \varphi_2^2 = 1$, which results in $\varphi_2 = \frac{\delta_{21}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}}$ and $\varphi_1 = \frac{\delta_{11}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}}$, as the normalized linear vector $\mathbf{\Phi}$.

General Multi-Component Functions

The insights gained from studying linear vectors can help in understanding the significance of the Karhunen-Loève transform. As demonstrated above, the transform amounts to solving a familiar eigenvector equation to determine the fundamental eigenvectors in the new transform domain.

Detection of linear vectors is the first step in decomposing a general function $g(x, t)$ into its fundamental functions. Linear trends often obscure data; for example, linear trends appear falsely as a low frequency component using Fourier transform techniques. Typically, pre-processing of the data is necessary to remove the linear trends by means of linear regression techniques, as shown in the fractal-wavelet technique. Using the Karhunen-Loève tranform, we aim to develop a unified decomposition method that will decompose the measured data into all of its fundamental components, including linear trends, sinusoidal patterns, stochastic patterns, and nonstationary patterns.

In this light, we next demonstrate the ability of the Karhunen-Loève transform to decompose a multicomponent signal. In order to extend the proof to the N-dimensional case, we present the following proofs in the continuous domain. Using a continuous function $x(t)$ composed of a linear function and a sinusoidal function, we assume that the manufacturing signal is represented with $x(t) = t + \sin(wt)$. The Karhunen-Loève decomposition is expected to result in a linear function and a sinusoidal function in the form of separate eigenfunctions.

In the continuous domain, the eigenvector Eq. (1.23) for the discrete case is converted to the following integral equation:

$$\int_0^T R(t_1, t_2)\Phi(t_2)dt_2 = \lambda\Phi(t_1) \tag{1.30}$$

where $R(t_1, t_2)$ is the covariance function for $x(t)$ and is defined as:

$$R(t_1,t_2) = E[x(t_1)x^*(t_2)]. \tag{1.31}$$

We need to solve the integral equation to determine the shape of the eigenfunctions $\phi_i\,(t)$, given the input function $x(t)$.

For a real input function $x(t) = t + \sin(wt)$, the covariance function becomes:

$$\begin{aligned} R(t_1,t_2) &= E[x(t_1)x(t_2)] = E[(t_1 + \sin(wt_1))(t_2 + \sin(wt_2))] \\ &= E[t_1t_2 + t_1\sin(wt_2) + t_2\sin(wt_1) + \sin(wt_1)\sin(wt_2)] \\ &= t_1t_2 + t_1\sin(wt_2) + t_2\sin(wt_1) + \sin(wt_1)\sin(wt_2). \end{aligned} \tag{1.32}$$

The expected value operator disappears since the signal is deterministic in this case. The integral equation now becomes:

$$\int_0^T (t_1t_2 + t_1\sin(wt_2) + t_2\sin(wt_1) + \sin(wt_1)\sin(wt_2))\phi(t_2)dt_2 = \lambda\phi(t_1). \tag{1.33}$$

The integral equation can be solved by differentiating it with respect to t_1 twice and applying the appropriate boundary conditions. The differentiation is performed using the Leibnitz Rule [78]. Differentiating the integral equation twice with respect to t_1, we obtain:

$$\int_0^T (-w^2t_2\sin(wt_1) - w^2\sin(wt_1)\sin(wt_2))\phi(t_2)dt_2 = \lambda\phi''(t_1). \tag{1.34}$$

Adding and subtracting $w^2\int_0^T(t_1t_2 + t_1\sin(wt_2))\phi(t_2)dt_2$, the equation becomes:

$$\begin{aligned} &-w^2\left(\int_0^T (t_1t_2 + t_1\sin(wt_2) + t_2\sin(wt_1) + \sin(wt_1)\sin(wt_2))\phi(t_2)dt_2\right) \\ &\quad + w^2\int_0^T (t_1t_2 + t_1\sin(wt_2))\phi(t_2)dt_2 = \lambda\phi''(t_1) \\ &\Leftrightarrow \lambda\phi''(t_1) + w^2\lambda\phi(t_1) = w^2\int_0^T (t_1t_2 + t_1\sin(wt_2))\phi(t_2)dt_2. \end{aligned} \tag{1.35}$$

Differentiating both sides of this relationship twice with respect to t_1 once again, we obtain the following differential equation:

$$\frac{d}{d^2 t_1}(\lambda\phi''(t_1) + w^2\lambda\phi(t_1)) = \lambda\phi''''(t_1) + w^2\lambda\phi''(t_1) = 0. \tag{1.36}$$

The fourth order differential equation is solved by first taking the Laplace transform of both sides, and then solving the characteristic equation for its roots [78], as follows:

$$s^4 + w^2 s^2 = 0 \Leftrightarrow s_1, s_2 = 0; \quad s_3, s_4 = +-jw. \tag{1.37}$$

The characteristic equation has two repeated roots, and two complex conjugate roots, resulting in the following general solution for $\phi_i(t)$:

$$\phi_1(t) = A + Bt \tag{1.38}$$

$$\phi_2(t) = C\cos(wt) + D\sin(wt). \tag{1.39}$$

Applying the boundary conditions $\phi_i(t_1 = 0) = 0$ and $\phi_i''(t_1 = 0) = 0$ to the general solution, we obtain $A = C = 0$, which results in the following eigenfunctions:

$$\phi_1(t) = Bt \tag{1.40}$$

$$\phi_2(t) = D\sin(wt). \tag{1.41}$$

The resulting eigenfunctions correspond to a linear function and a sinusoidal function, respectively. Each of these functions is a solution to the differential equation.

Monitoring Decomposed Functions in Signals

As demonstrated above, the Karhunen-Loève transform offers the ability to decompose a given signal into its fundamental components. This ability is present whether the signal's individual components are deterministic (e.g., linear and sinusoidal functions), stochastic, stationary, or nonstationary. The ability of the Karhunen-Loeve transform to decompose different types of signals will be demonstrated in a later section with numerically-generated signals which simulate faulty signals from manufacturing [69, 70].

In this work, we claim that monitoring for faults in manufacturing is accomplished by means of the coefficient vectors corresponding to each fundamental eigenfunction. In this section, we demonstrate this property with a simple example. The coefficient vectors will be revisited in a later section using numerically-generated signals.

To demonstrate the use of coefficient vectors for monitoring changes in the fundamental eigenfunctions, let us briefly revisit the simple case of two linear vectors. Let the coefficient vector $Y = [y_1 y_2]^T$ corresponding to the fundamental eigenvector $\Phi = [\varphi_1 \varphi_2]^T$ indicate the change in the slope between the two input vectors X_1 and X_2 [70]. The coefficient vector $Y = [y_1\ y_2]^T$ corresponding to the eigenvector $\Phi = [\varphi_1 \varphi_2]^T = \left[\frac{\delta_{11}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} \frac{\delta_{21}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}}\right]^T$ is computed using $Y = \Phi^T Z$, which becomes:

$$\begin{bmatrix} y_1 & y_2 \end{bmatrix} = \begin{bmatrix} \frac{\delta_{11}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} & \frac{\delta_{21}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} \end{bmatrix} \begin{bmatrix} -\delta_{11} & \delta_{11} \\ -\delta_{21} & \delta_{21} \end{bmatrix} = \begin{bmatrix} -\sqrt{\delta_{11}^2 + \delta_{21}^2} & \sqrt{\delta_{11}^2 + \delta_{21}^2} \end{bmatrix}. \tag{1.42}$$

The change between the two input vectors is equal to the difference $Z_2 - Z_1$, which is equal to the sum of the δ_{ij} deviations from the mean vector:

$$Z_2 - Z_1 = \begin{bmatrix} \delta_{11} \\ \delta_{21} \end{bmatrix} - \begin{bmatrix} -\delta_{11} \\ -\delta_{21} \end{bmatrix} = \begin{bmatrix} 2\delta_{11} \\ 2\delta_{21} \end{bmatrix}. \tag{1.43}$$

Since $X_i = Z_i + X_a$, we can see that $X_2 - X_1 = Z_2 - Z_1$, which indicates the change between the two input vectors.

Predicting Fault Patterns in Signals

To demonstrate the ability of the Karhunen-Loève method to obtain a reconstruction of the original signal for error prediction purposes, we again revisit the simple case of linear vectors. The reconstruction ability of the method will be demonstrated further in a later section using numerically-generated signals [69, 70]. The original data vectors X_1, and X_2 can be reconstructed as a linear combination of the eigenvector $\Phi = [\varphi_1 \varphi_2]^T$ and the coefficient vector $Y = [y_1\ y_2]^T$ in the transform domain. The zero-mean data matrices Z_1, and Z_2 are reconstructed using $Z_1, = y_1\Phi$ and $Z_2 = y_2\Phi$, which become:

$$Z_1 = -\left(\sqrt{\delta_{11}^2 + \delta_{21}^2}\right) \begin{bmatrix} \dfrac{\delta_{11}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} \\ \dfrac{\delta_{21}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} \end{bmatrix} = \begin{bmatrix} -\delta_{11} \\ -\delta_{21} \end{bmatrix} \tag{1.44}$$

$$Z_2 = \left(\sqrt{\delta_{11}^2 + \delta_{21}^2}\right) \begin{bmatrix} \dfrac{\delta_{11}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} \\ \dfrac{\delta_{21}}{\sqrt{\delta_{11}^2 + \delta_{21}^2}} \end{bmatrix} = \begin{bmatrix} \delta_{11} \\ \delta_{21} \end{bmatrix}. \tag{1.45}$$

When the mean vector X_a is added to the zero-mean vectors Z_1 and Z_2, we obtain the original data vectors X_1 and X_2.

Monitoring and Prediction of Numerically Generated Surface Signals

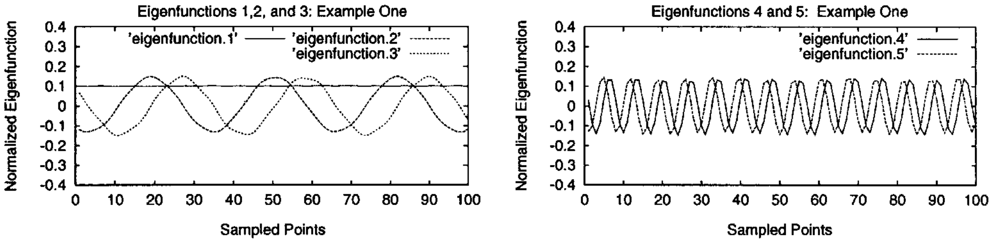
Up to this point, we have discussed the potential uses of the Karhunen-Loève transform and presented the mathematical extension to demonstrate its mechanics. It is now necessary to show that the method can be used effectively with actual signals. In this light, the purpose of this section is to demonstrate the ability of the Karhunen-Loève method to decompose complex signals into physically meaningful patterns which can be monitored individually. For this purpose, we present two examples using numerically-generated signals to simulate manufacturing signals [69].

The first example is the case of a multi-component signal, composed of three deterministic components: a linear function and two sinusoidal functions with different amplitudes and frequencies. From this study, we expect two things: (1) the multi-component signal can be decomposed into a few fundamental patterns, which can be used to reconstruct an estimate of the original signals; and (2) the multi-component signal can be decomposed into individual eigenfunctions which can be monitored separately by means of the corresponding coefficient vectors.

The second example investigates the case where nonstationary changes occur in the signal of the first example, with added noise. These changes include: (1) a change in the mean-square value simulated by a sudden change in the amplitude of one of the sinusoidal components; and (2) a change in the mean

TABLE 1.11 Dominant Eigenvalues for Examples 1 and 2

Index	Example 1	Example 2
1	1873.525	555.528
2	103.499	283.348
3	95.351	259.521
4	25.411	25.997
5	24.183	25.322



(a) Eigenfunctions #1, #2, and #3.

(b) Eigenfunctions #4 and #5.

FIGURE 1.11 Decomposition of profiles from Example 1.

value simulated by a sudden change in the offset level of the signal. We expect to decompose the signal into fundamental eigenfunctions, and monitor their corresponding coefficient vectors. The coefficient vectors must indicate where the nonstationary change occurred and what the characteristics of the nonstationary change are. Finally, an estimate of the original signals will be reconstructed to demonstrate that the effect of added noise is reduced.

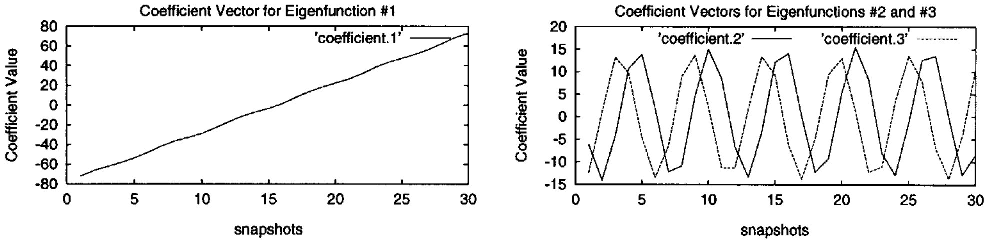
We assume that snapshots of the manufacturing process are collected on a regular basis, allowing the continual monitoring of the process. We expect to use these snapshots to detect the occurance of faults. M represents the number of snapshots collected, N is the number of points sampled per snapshot. Table 1.11 presents the resulting dominant eigenvalues for the two cases.

Example One: Decomposition of Multi-Component Signals

The first example involves a multi-component signal composed of two sinusoidal patterns and a linear pattern [69]. When standard Fourier-based techniques are used to analyze data, linear trends typically result in misleading information. As a result, linear patterns are typically removed prior to analyzing signal characteristics. However, the temporal or spatial occurrence of linear trends can provide valuable information about the status of the manufacturing process.

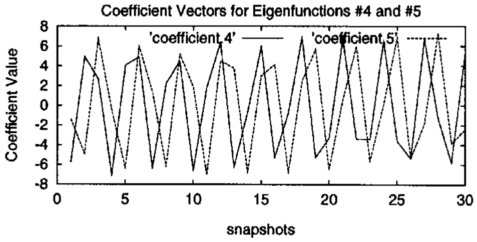
In this example, we generate numerical signals simulating a manufacturing signal composed of two sinusoidal functions and linear function. The multi-component signal has the form of $A_1\sin(F_{1j}) + A_2\sin(F_{2j}) + B_{1j} + B_2$: two sinusoidal components with one high frequency ($F_1 = 0.9$ rad/sec), small amplitude ($A_1 = 1$ mm), and the other low frequency ($F_2 = 0.2$ rad/sec) and large amplitude ($A_2 = 2$ mm), and a linear component with slope $B_1 = 0.005$ and y-intercept $B_2 = 0.0$. We want to decompose the multi-component signal into eigenfunctions representing a linear trend and two sinusoidal functions. The eigenfunctions can then be monitored individually to detect potential changes in the manufacturing system. $M = 30$ snapshots of the manufacturing process are assumed to be collected; each shapshot contains $N = 100$ sampled points.

The Karhunen-Loève decomposition of the data in this example results in five fundamental eigenfunctions, as shown in Fig. 1.11. The corresponding eigenvalues are shown in Table 1.11. The first eigenfunction corresponds to a function representing a linear trend. The second and third eigenfunctions correspond to the low-frequency sinusoidal component. Notice than, instead of one eigenfunction, we obtain two



(a) Coefficient Vector #1.

(b) Coefficient Vectors #2 and #3.



(c) Coefficient Vectors #4 and #5.

FIGURE 1.12 Coefficient vectors for Example 1.

eigenfunctions which look alike, except for a phase shift. This is due to the fact that the phase angle of each snapshot is different, hence introducing a second eigenfunction [70]. Recall that phase information is typically lost with Fourier-based methods. Similarly, the fourth and fifth eigenfunctions correspond to the high-frequency sinusoidal component, and the corresponding phase-shifted version. Note that each one of these eigenfunctions can be studied individually to understand the nature of the fault patterns in a complex signal. Sinusoidal components are isolated without any misinterpretation caused by the obscuring of the data by other components. The linear trend is separated from the sinusoidal components. For example, if a third frequency component is introduced, it would appear as an additional eigenfunction pair. This might, for example, correspond to the appearance of harmonics due to a bearing misalignment in the manufacturing machine. Its immediate identification is crucial for the diagnosis of the fault mechanism.

The coefficient vectors corresponding to each of the fundamental eigenfunctions are shown in Fig. 1.12. Note that the first coefficient vector shows the change in slope of the linear trend isolated in the form of eigenfunction #1 (Fig. 1.11). Using the eigenfunction and the corresponding coefficient vector, the introduction of linear trends can be monitored separately from the rest of the significant components. In addition, changes in the slope of existing linear trends can be monitored as well. The introduction of slopes and/or changes in the slope of existing linear trends may be indicative of serious errors of form in the system. Their accurate identification is crucial in identifying and eliminating the origin of the errors. The remaining coefficient vectors correspond to the values of each sinusoidal eigenfunction along the snapshots. This can be viewed as the value of each snapshot projected onto the eigenfunctions in the new transform domain.

It is important to note that the Karhunen-Loève transform results in $M = 30$ eigenfunctions. The dimensionality in the new domain is determined by the rank of the covariance matrix, which is determined by the number of input snapshots. As shown in Table 1.11, only a subset ($M = 5$) of these eigenfunctions is significant, containing the majority of the total energy. As expected, the reconstruction of the original profiles using these five eigenfunctions is perfect, as shown in Fig. 1.13. Recall that the

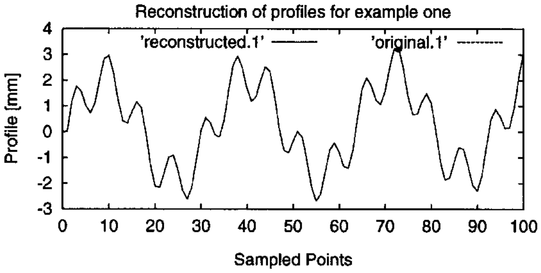


FIGURE 1.13 Reconstruction of profiles for Example 1.

ability of the Karhunen-Loève method to reconstruct accurate estimates of the original signals is an important requirement to allow error prediction. In theory, if we know the eigenfunctions resulting from the manufacturing machine and process conditions, then we can assume a set of coefficient vectors given the machining and part specifications, and predict the nature of the expected surface profiles. We could also reconstruct without the linear component, for example, to determine the resulting surface profiles once the first error source has been eliminated. The ability to reconstruct estimates of surface profiles opens up avenues for tolerance assignment, machine design, and process design, prior to manufacturing a part.

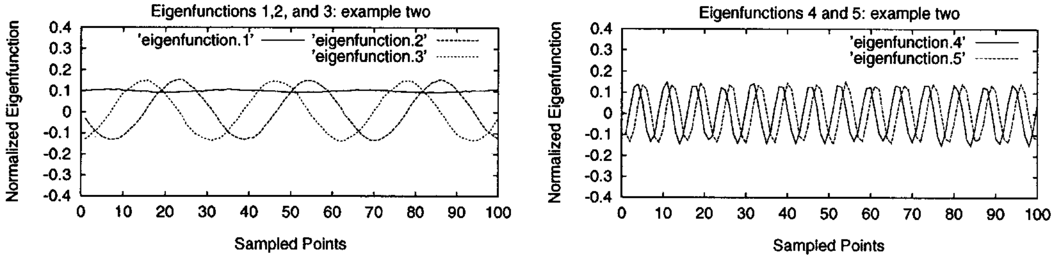
Example Two: Detection of Nonstationary Patterns

The second example involves a multi-component signal with random noise, in which two nonstationary changes occur: (1) a change in the magnitude of one of the sinusoidal patterns; (2) a change in the offset level [69]. Nonstationarities typically obscure the true nature of data and result in misleading information about the data. In addition, the nature of the nonstationarities themselves are often difficult to determine; this can provide valuable information about potential failures in the manufacturing machine.

In this example, we generate signals to simulate the occurrence of faulty patterns during the normal operation of a manufacturing process. A total of $m = 1 \cdots M$, $M = 60$, snapshots are assumed to be collected from the manufacturing process, each with $N = 100$ sampled points. The first $m = 10$ snapshots correspond to the case of two sinusoids, combined with random noise. These represent the normal (faultless) conditions of the manufacturing signal. Then, we introduce a nonstationary change in the mean-square value of the signal: the amplitude of one of the sinusoidal components (low-frequency component) increases suddenly for the next $m = 10$ snapshots, and then returns to its normal status. The second frequency component remains unchanged. Finally, a second nonstationary change in the mean value occurs at the $m = 30$th location in time (or space): the offset level of the signal jumps by five units for the next $m = 10$ snapshots. From $m = 1 \cdots 20$, the multi-component signal has the form of $A_1 \sin(F_{1j}) + A_2 \sin(F_{2j})$ ($F_1 = 0.9$ rad/sec, $A_1 = 1$ mm, $F_2 = 0.2$ rad/sec, and $A_2 = 2$ mm), plus random (Gaussian) noise (zero mean and variance of 0.09).

From $m = 21 \cdots 40$, the signal has the same form, but the amplitude of the low-frequency sinusoidal component has increased to $A_2 = 5$ mm. Finally, from $m = 41 \cdots 60$, the sinusoidal component has returned to its original state, but instead, there is an offset change of $OFF = 5.0$ added to the original signal. The complexity of this scenario typically prevents the combination of these nonstationary changes from being detected accurately in nature and in time (or space). We show here that the Karhunen-Loève method transforms the collected data into fundamental components which can be monitored individually to isolate both types of changes discussed.

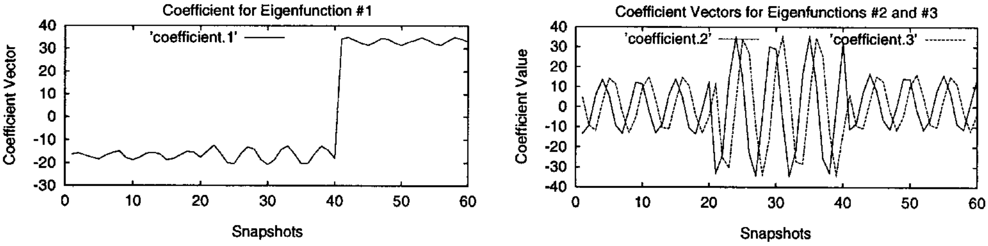
The Karhunen-Loève decomposition of data in the second example also results in five fundamental eigenfunctions, as shown in Fig. 1.14. The corresponding eigenvalues are shown in Table 1.11. Once again, the decomposition results in a linear eigenfunction, and two pairs of eigenfunctions corresponding to the two sinusoidal components. The first eigenfunction represents the fundamental pattern due to the offset



(a) Eigenfunctions #1, #2, and #3.

(b) Eigenfunctions #4 and #5.

FIGURE 1.14 Decomposition of signals from Example 2.



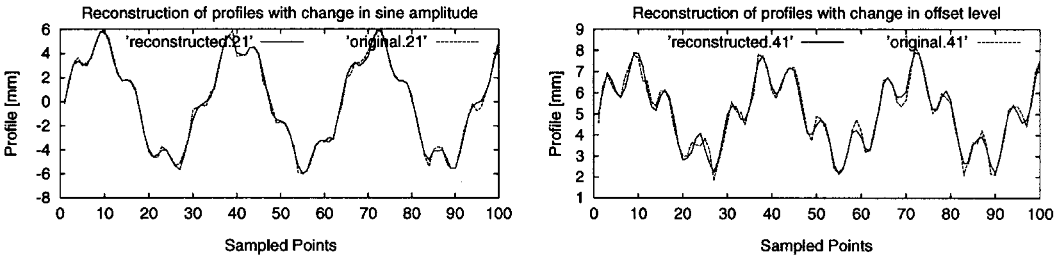
(a) Coefficient Vector #1.

(b) Coefficient Vectors #2 and #3.

FIGURE 1.15 Coefficient vectors for Example 2.

change introduced in the multi-component signal. Note that the eigenvalue for the first eigenfunction is not as significant as the first eigenvalue in the first example. The eigenvalue is related to the variance of the coefficients, and hence is greater in the case of a linear trend with increasing slope. Note that the eigenvalues for the second and third eigenfunctions (corresponding to the low-frequency sinusoidal component) are higher than in the first example. This is due to the fact that the amplitude of the low-frequency sinusoidal component increases during the monitoring process. This nonstationary change is reflected on the eigenvalues since they are computed based on the coefficient vectors. Recall that the coefficient vectors should reflect the change in the amplitude of the low-frequency sinusoidal component. Notice that the shape of the first pair of sinusoidal eigenfunctions remains the same as in the first example, since the fundamental characteristics (e.g., frequency and phase) of the sinusoidal components do not change. Finally, the eigenvalues and shape for the second pair of eigenfunctions (corresponding to the high-frequency sinusoidal component) remain the same as in the first example, since no change is introduced to this component. The slight change in the eigenvalues (#4 and #5) is due to the presence of the noise component in the second example.

The coefficient vectors corresponding to the five fundamental eigenfunctions are shown in Fig. 1.15. As expected, the coefficient vector corresponding to the linear eigenfunction (#l) indicates the change in the offset level starting from the snapshot $m = 40$. This is a crucial result, since it provides us with the ability to determine the exact location and severity of the nonstationary fault pattern in the signal. Recall that it is often very difficult to determine these two properties about nonstationary changes in a signal. The coefficient vectors corresponding to low-frequency eigenfunctions (#2 and #3) indicate the change in amplitude between snapshots $m = 20$ and $m = 40$. This is also a crucial result, since such a nonstationary change would be typically averaged out using Fourier-based techniques. In our case, by monitoring the eigenfunction and the corresponding coefficient function simultaneously, we are able to detect



(a) Signal with Mean-Square Change.

(b) Signal with Mean Value Change.

FIGURE 1.16 Reconstruction of signals in Example 2.

the location and severity of the nonstationary change in the signal. No change is indicated in the coefficient vectors corresponding the eigenfunctions #4 and #5, as expected.

Finally, the reconstruction of the profiles from different instances during monitoring are shown in Fig. 1.16. Figure 1.16(a) corresponds to a profile from the second set of snapshots where a nonstationary change in the mean-square value occurs ($m = 21$). Fig. 1.16(b) is a profile from the third set of snapshots where a nonstationary change in the mean value occurs ($m = 41$). Each profile is reconstructed using the same five significant eigenfunctions and their corresponding coefficient vectors, in the form of a linear combination. Notice that the reconstruction in this case is not as perfect as in the first example. This is due to the fact that the random component in the original signal has been reduced in significance during reconstruction, hence increasing the signal-to-noise ratio. The majority of the stochastic effect is concentrated in the lower-eigenvalue components after the Karhunen-Loève decomposition, resulting in the non-significant eigenfunctions. Once again, the ability to reconstruct every instant of a manufacturing process, with or without nonstationary fault patterns, is a crucial requirement in error prediction for *a priori* tolerance assignment and machine and process redesign.

Understanding and Redesigning a Manufacturing Process through Surface Analysis

To demonstrate that the Karhunen-Loève method can be used with real data from manufactured surfaces, in this section, we discuss the potential of the our technique in integrating the design and manufacturing cycles. The manufacturing process under study is Selective Laser Sintering [68]. Selective Laser Sintering is a layered-manufacturing process where thin layers of powder are successively scanned to shape by means of a laser and galvanometer mirrors. Roller and piston mechanisms are used to deposit and spread each layer of powder on a powder bed.

One of the main concerns in this layered-manufacturing process is the ability to obtain accurate parts on a repeatable basis. Any of the components of the Selective Laser Sintering machine may result in undesirable fault patterns on part surfaces. As a result, it is important to understand the surface characteristics of parts produced using this process [68]. Surface characteristics provide a "fingerprint" of the manufacturing process under monitoring. Consequently, the understanding gained from the surface information will help in the understanding of the manufacturing process and its potential limitations.

What we offer in this work is a means to systematically analyze and synthesize the fault patterns on part surfaces. The idea is to provide the manufacturers with an accurate means to communicate the fault status to the designers [69, 70]. The designers can then take remedial actions to eliminate the problems. Remedial actions may include redesign of the manufacturing machine components and parameters, and/or control of the process parameters. In this fashion, we achieve two goals: (1) provide a tool to close the loop between design and manufacturing; and (2) provide a tool to understand new manufacturing processes and machines.

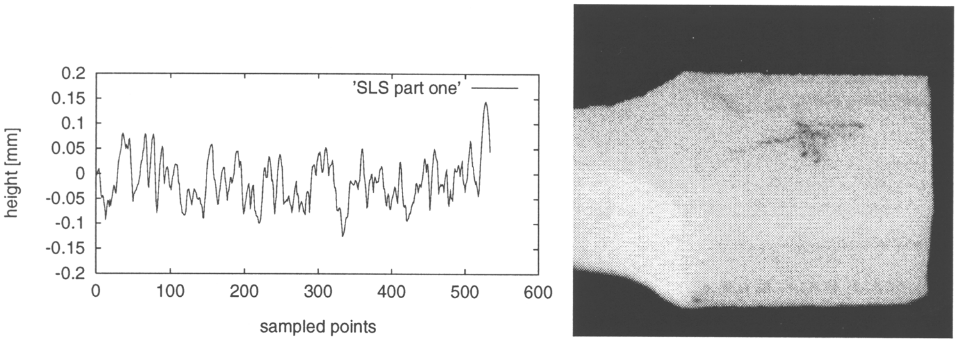


FIGURE 1.17 Surface analysis of Part #1.

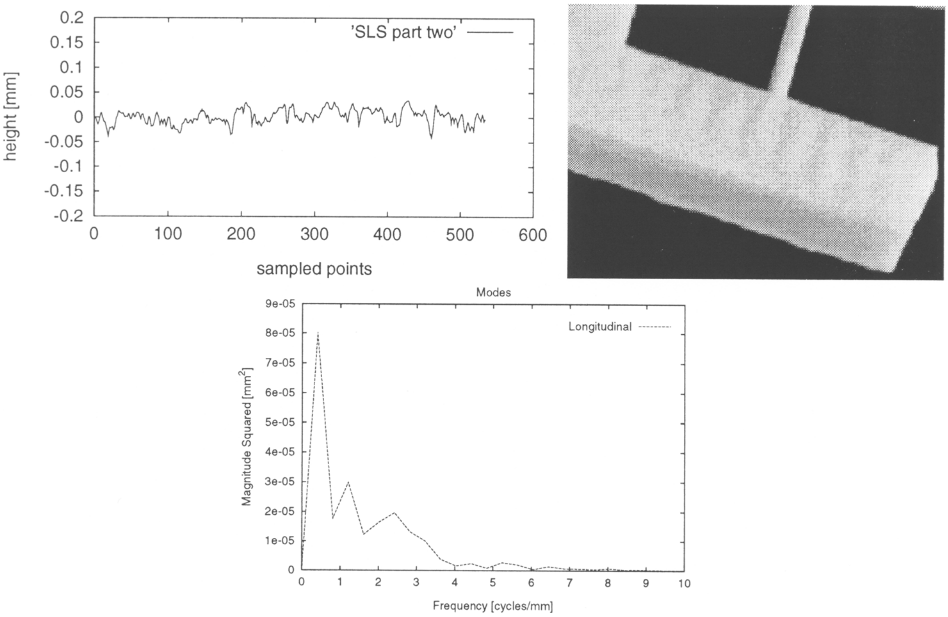


FIGURE 1.18 Surface analysis of Part #2.

To demonstrate these goals, we first investigate surfaces of parts built using a polycarbonate powder [68]. The surface profile of such a part is shown in Fig. 1.17. These surfaces indicate a rough structure, which the designers would like to eliminate. The rough structure is a consequence of the material choice, and is unacceptable for customer satisfaction. Additional finishing processes typically follow part production to eliminate this problem. Analysis of these surface profiles using the Karhunen-Loève transform reveals a predominantly stochastic structure, without any coherent structures revealed. However, the roughness of the surfaces is in the order of 30 m, which is the same order of magnitude as the particle diameter of the powder used to build this part. This information is passed onto the designers. The designer's recommendation is to change the powder type and particle size to remedy the problem.

Figure 1.18 shows a part produced using a powder with smaller particle size. The corresponding surface profile shows an improvement over the previous part profile. In fact, the surface roughness has been reduced to an average value of 10 m, and the part surface is no longer rough to touch. The improvement on the surface roughness is satisfactory to the designer and the customer. However, as a consequence of the first remedial action, new fault patterns are revealed on the new part surfaces. We first use a standard

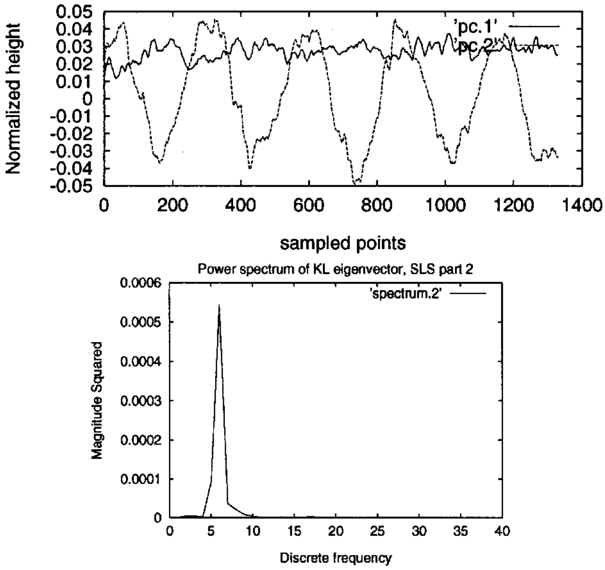


FIGURE 1.19 Karhunen-Loève analysis of surfaces from Part #2.

surface analysis tool: a Fourier transform analysis of the surface profile reveals a structure which is no longer random in nature (Fig. 1.18); however, the distinction in frequency components is not clear using the Fourier-based analysis. Remedial actions by designers can only be taken if the information received from the manufacturers is clear and unambiguous.

To obtain a clearer picture, we next analyze the same surface (Fig. 1.18) using the Karhunen-Loève transform technique. This analysis results in two eigenvectors, which are identified as the two dominant components, shown in Fig. 1.19. The sinusoidal component is clear and distinct, as shown with the accompanying Fourier transform. This provides an unambiguous way of detecting the correct frequency component corresponding to the low-frequency waveform on part surfaces. The waveform corresponds to the roller chatter frequency identified from vibrational measurements [68]. By collecting vibrational measurements from the roller mechanism, the designer can match the waveform frequency to the roller chatter frequency. Monitoring both submechanisms will provide a means to assure that the magnitude of the surface fault pattern is below an unacceptable threshold. The threshold levels can be set by the customer specifications. When this threshold is exceeded, the fault pattern is deemed severe; the remedial action taken by the designer is to look into ways of redesigning the roller mechanism. This systematic surface analysis procedure provides a means to understand the potential fault mechanisms in the manufacturing process under investigation. This information is conveyed to the designer, who then implements remedial measures to prevent or eliminate the potential problems. This analysis can be carried out for different surfaces to determine and categorize the different fault mechanisms affecting surface quality during part production [68].

Finally, to investigate the potential of *a priori* prediction of the fault mechanisms, we compute an estimate of the surface patterns on the Selective Laser Sintering parts. The reconstruction is shown in Fig. 1.20, as compared to the original profile measurement. The reconstructed profile compares to the original profile fairly closely. The designers can use such tools to predict what the fault patterns will look like, given the manufacturing specifications. For example, if the surface roughness is the important criterion for a satisfactory profile, designers may decide to build the part as is, without eliminating the low-frequency waveform identified earlier.

This simple design example using a real industrial problem illustrates the potential of the Karhunen-Loève analysis technique in integrating the design and manufacturing phases. The potential of such an implementation is invaluable for our idealistic vision of automated design and manufacturing. Automated

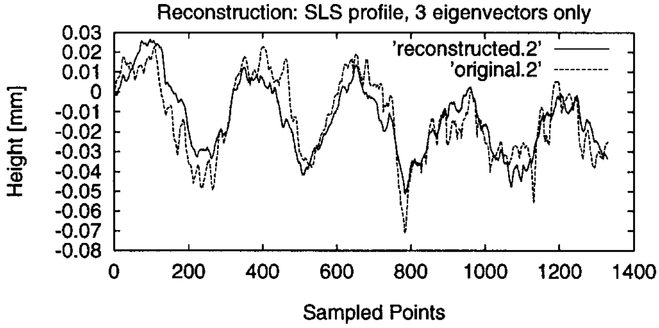


FIGURE 1.20 Karhunen-Loève synthesis of surfaces from Part #2.

design and manufacturing implies the implicit goal of being able to manufacture fault-free products. Fault-free part production benefits tremendously from a thorough understanding of the potential fault mechanisms during any manufacturing process.

1.5 Discussion: Extensions and Ending Thoughts

Our short-term vision of surface analysis and synthesis methods is nearly complete. We have constructed an approach to separate the trend, periodic, fractal, and outlier components of a surface. These components are then represented and combined with a unique wavelet and superposition model. The result provides the capability to understand and analyze structure in stochastic aspects of manufacturing processes, impacting tolerance and surface finish analyses. It also provides us with a means to synthesize representative surfaces of manufacturing processes, as a function of both process and design parameters.

To complement this approach, we have also significantly extended the Karhunen-Loève transform, with applications to surface analysis. The presence of time varying effects can invalidate the results of the superposition representation. A large number of deterministic components, in conjunction with noisy signals, can also lead to inaccuracies. The extended Karhunen-Loève transform overcomes these issues. It decomposes a signal into its fundamental components, even in the presence of large magnitude noise. Nonstationarities are also separated as distinct components, providing a robust representation of the primary terms of the superposition model, besides the fractal component.

Dichotomy or Integration?

While the described methods provide significant advancements to surface representation, a fundamental question arises: are the two methods distinct, or do they naturally combine to provide a unified method? Our vision, as stated upfront, is to provide a unified approach. We want an approach that does not assume a certain type of signal, but instead can represent general signal components without *a priori* knowledge of the generating dynamics. We also want an approach that handles a spectrum of noise components, as well as nonlinear structure (fractals).

A careful study of the fractal approach and Karhunen-Loève transform shows that they are not capable, in and of themselves, to satisfy this vision. Both have some deficiencies. The fractal method is not able to represent certain nonstationarities, and it has difficulties with large numbers of deterministic components. Likewise, the extended Karhunen-Loève transform separates the stochastic component without distinguishing the possible structure in that component.

Given their strengths and deficiencies, it is natural to unify the methods to satisfy our vision for surface analysis and synthesis. The superposition model is the key element for this unification. The Karhunen-Loève method is robust at representing and monitoring the deterministic components of a surface signal, thus replacing those components in the model. The fractal method using wavelets, on the other hand, providing a means of analyzing the differential stochastic component from the Karhunen-Loève analysis.

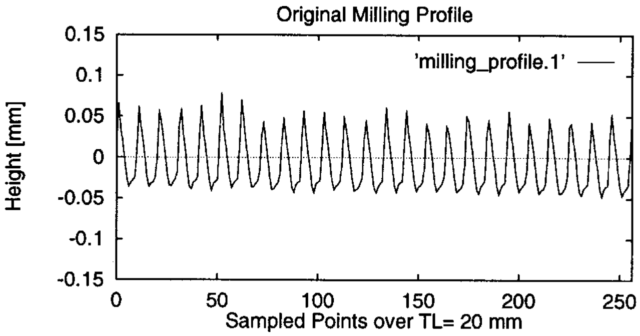


FIGURE 1.21 A sample milling profile.

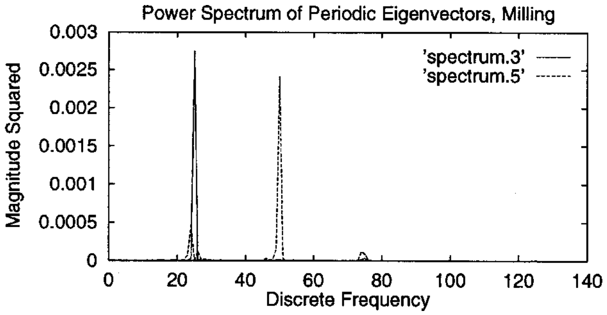


FIGURE 1.22 Power spectra of the dominant KL eigenvectors.

In the next section, we expound on this relationship, illustrating the unification with a design and manufacturing example.

Conjunction of the Two Methods

To perform the proposed conjunction, we investigate the surface of a part generated using the milling process described earlier in the chapter. In the following, the Karhunen-Loève technique is first used to isolate and characterize the deterministic components in the milling surface profiles. Next, the milling profiles are filtered by removing the fundamental KL eigenvectors. This step is followed by the application of the fractal-wavelet technique to the filtered surfaces to represent the characteristics of the remaining stochastic component.

We first analyze these surfaces using the Karhunen-Loève transform. $M = 20$ measurements of such a milling surface are collected over a 20 mm total traverse length, each containing $N = 256$ data points. The Karhunen-Loève transform results in five fundamental eigenvectors, adding up to 93 percent of the total energy. A sample profile is shown in Fig. 1.21.

The first eigenvector corresponds to the dominant frequency component and is identified clearly using the transform. This component has the same frequency as the one identified earlier in the fractal-wavelet analysis of the milling profiles. While the first dominant eigenvector captures the dominant characteristics of the milling profiles, the KL analysis reveals additional modes which were not detected with the fractal-wavelet technique. To confirm these results, we take advantage of the standard Fourier transform for purely periodic signals: Fig. 1.22 shows the power spectrum of the eigenfunctions, indicating the main frequency component at discrete frequency $l = 25$, and two additional frequency components, at discrete frequencies $l = 50$ and $l = 75$. The first frequency component, identified with the first eigenvector, is the component corresponding to the feed marks during the milling process. The second and third frequency components are identified as harmonics of the first component, due to the tool vibrations

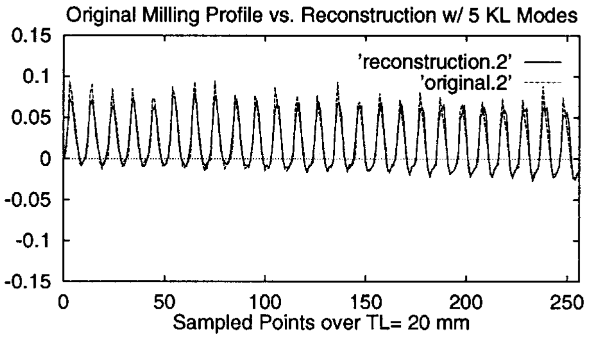


FIGURE 1.23 Original milling profile vs. KL-reconstructed milling profile.

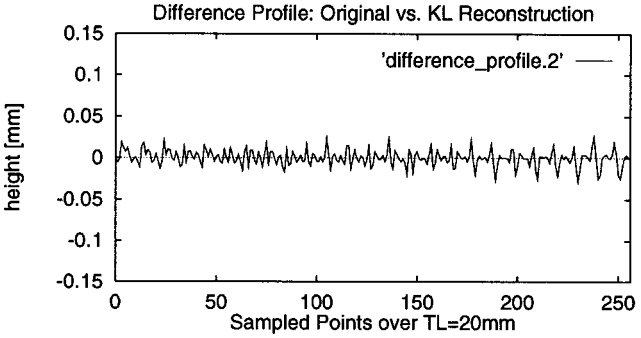


FIGURE 1.24 KL-filtered milling profile.

during milling. Recall that, since the eigenfunctions have normalized magnitudes, the magnitude of the eigenfunctions does not represent their significance in the milling profiles. Each of these frequency components can be monitored by means of the corresponding coefficient vectors to determine their severity and potential harm to the manufacturing equipment and product.

Reconstruction using the first five fundamental eigenvectors results in the representation of the main deterministic patterns in the milling profiles. An example is shown in Fig. 1.23, where a reconstructed deterministic profile is compared to an original milling profile. The reconstruction compares to the original profile closely. However, the stochastic information is not contained in these five eigenvectors, and hence monitoring of the stochastic component is not possible with the Karhunen-Loève technique.

Let us next look at the milling profiles using the fractal-wavelet technique. Figure 1.24 shows the difference between an original milling profile and the KL reconstruction of the same profile using five eigenvectors. The difference profile has been "filtered" by removing the fundamental deterministic modes. The fractal-wavelet technique is used on this filtered profile, resulting in a fractal dimension of 1.86. This fractal dimension compares favorably to the milling fractal results presented in Table 1.5. As a result, the stochastic component of the milling profiles can be effectively monitored using the fractal measures. Furthermore, the conjunction of the two techniques using a superposition approach is validated with these results.

This simple example presents one way of combining the two novel techniques introduced in this chapter. Such a conjunction brings us one step closer to the idealistic vision presented in this chapter. The Karhunen-Loève transform detects the dominant deterministic modes and nonstationary patterns in manufacturing signals. The method provides an effective means of monitoring, representing, and diagnosing the condition of surfaces from precision manufacturing processes, in the form of fundamental eigenvectors and coefficient vectors. Then, the fractal-wavelet method is used following the Karhunen-Loève step to measure and monitor the condition of the stochastic portion of the manufacturing surfaces by means of fractal measures. In this fashion, the proposed conjunction provides the manufacturers and designers with a unified method to detect, represent, and monitor different types of fault patterns during manufacturing.

Future Implications

The vision of surface analysis engendered in this chapter literally builds on the shoulders of the veterans in the field [14, 25, 39, 43, 63, 74]. Their keen insight and perception changed the way we practice surface analysis. Ad hoc methods are no longer the norm. Instead, mathematics and formal techniques are offered to monitor and evolve the manufacturing of products.

We hope that our additions have enriched these insights. Clearly, the ability to fingerprint manufactured surfaces, even if they combine determinism, nonstationarities, and stochastic components, is a step forward. The same is true for simplifying and devoting equal weight to surface synthesis. The foundation then exists for building stable bridges between manufacturing and design.

But only the tip of the iceberg has been breached with our additions. Many avenues exist for further advancements. As a consequence of our vision, three such avenues are paramount: linking design and manufacturing more closely with desktop tools for tolerance analysis and synthesis; real-time monitoring and diagnosis of manufacturing process for machine precision and redesign; and closing the loop in manufacturing with process control. For the first of these, tolerance analysis still remains a nebulaus task in most industries. Besides tools for assembly tolerances, few mathematical techniques exist at the fingertips of the designer. Even if these tools did exist, concrete data and fingerprinting of the available manufacturing processes are nonexistent (except for average measures or cumulative SPC). Computational systems are thus needed that will provide functional tolerancing capabilities through surface synthesis, tapping into the fingerprinted processes. The designer will then have a means, potentially, to vary design and process parameters earlier in the product development process.

The second avenue, real-time monitoring and diagnosis, is the antithesis of function tolerancing. The goals here are to evolve the precision of manufacturing equipment. Advanced monitoring with methods such as Karhunen-Loève will provide the capability to detect and avoid low precision envelopes in manufacturing. Likewise, diagnostics will provide pointers to subsystems that require technological advancement and redesign. However, there will be resistance to implementing new techniques on the manufacturing floor, for obvious reasons. Advancements of simple tools and interfaces that build on the more fundamental mathematics will be the keys to success.

A third paramount avenue entails process control. Our unified approach, or its equivalent, provides significant insight into the underlying physics of the manufacturing processes. These insights, if advanced, could provide new models for on-line control. The result would be a competitive alter-native to monitoring and diagnosis.

These future avenues will not be easy. Investment will be needed in time and resources to fingerprint the current manufacturing processes and equipment [81]. The resulting data will then need to be stored, maintained, and continually updated to monitor lifecycle issues and to assure the robustness of the fingerprints. These future avenues may also call for new approaches to metrology, computational techniques, and operator interfaces. Without such approaches, pragmatic implementation may not be possible.

While significant investment will certainly be needed, the potential payoff is dramatic. Imagine a new world of product development where designers are able to design for manufacturing more directly. In this idealistic future, manufacturing processes will be understood more fully and controlled more robustly. Designers and manufacturers will no longer dismiss or passively accept uncertainties in fabricating products, but provide vehicles to process and product improvement. This new world is the epitomy of bridging design and manufacturing, and it may be only an arm's reach away.

Acknowledgments

This material is based on work supported, in part, by The National Science Foundation, Grant No. DDM-9111372; an NSF Presidential Young Investigator Award; a grant from TARP; and grants from Ford Motor Company, Texas Instruments, and the June and Gene Gillis Endowed Faculty Fellowship in Manufacturing.

Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsors.

References

1. Ahmed, N. and Rao, K. R. *Orthogonal Transforms for Digital Signal Processing.* Springer-Verlag, New York, 1975.
2. Akansu, A. N. and Haddad, R. A. *Multiresolution Signal Decomposition: Transforms, Subbands, Wavelets.* Academic Press, Inc., San Diego, 1992.
3. Algazi, V. R., Brown, K. L., and Ready, M. J. Transform representation of the spectra of acoustic speech segments with applications, part I: General approach and application to speech recognition. *IEEE Transactions on Speech and Audio Processing,* 1(2):180, April 1993.
4. Ball, K. S., Sirovich, L., and Keefe, L. R. Dynamical eigenfunction decomposition of turbulent channel flow. *International Journal for Numerical Methods in Fluids,* 12:585, 1991.
5. Barker, R. W., Klutke, G., and Hinich, M. J. Monitoring rotating tool wear using higher-order spectral features. *Journal of Engineering for Industry,* 115:23, February 1993.
6. Barnett, V. and Lewis T. *Outliers in Statistical Data.* John Wiley & Sons, New York, 1978.
7. Bendat, J. S. and Piersol, A. G. *Random Data: Analysis and Measurement Procedures.* John Wiley & Sons, New York, 1986.
8. Berry, J. E. How to track rolling element bearing health with vibration signature analysis. *Sound and Vibration*, p. 24, November 1991.
9. Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. *Time Series Analysis: Forecasting and Control.* Prentice Hall, New Jersey, 1994.
10. Braun, S. *Mechanical Signature Analysis: Theory and Applications.* Academic Press, London, 1986.
11. Chui C. K. *An Introduction to Wavelets. Wavelet Analysis and its Applications.* Academic Press, Boston, 1992.
12. Cook, N. H. and Shaw, M. C. Discontinuous chip formation. *Transactions of the ASME,* 76:153, 1954.
13. Daubechies, I. The wavelet transform, time frequency localization, and signal analysis. *IEEE Transactions on Information Theory,* 36(5):961, September 1990.
14. DeVries, W. R. The surface typology of journals. *Annals of the CIRP,* 31(l):467, 1982.
15. Dong, W. and Joe Au, Y. H. Machine tool chatter monitoring by coherence analysis. *International Journal of Production Research,* 30(8):1901, 1992.
16. Fackrell, J. W. A., White, P. R., and Hammond, J. K. Bispectral analysis of periodic signals in noise: theory, interpretation, and condition monitoring. In *EUSIPCO'94,* Edinburgh, UK, 1994.
17. Farmer, D. A., Brecker, J. N., and Shaw, M. C. Study of the finish produced in surface grinding: Part 1—experimental. *Proceedings of the Institution of Mechanical Engineers,* 182:171, 1967.
18. Freund, L. B. *Dynamic Fracture Mechanics.* Cambridge University Press, Cambridge, 1990.
19. Fukunaga K. *Introduction to Statistical Pattern Recognition.* Academic Press, New York, 1972.
20. Gadhok, J. S. and Zychowski, E. M. Quantizing laser scanner performance by signature analysis. *SPIE: Laser Scanning and Recording,* 498:45, 1984.
21. Gagnepain J. J. and Roques-Carmes, C. Fractal approach to two-dimensional and three-dimensional surface roughness. *Wear,* 109:119, May 1986.
22. Geng, Z. and Qu L. Vibrational diagnosis of machine parts using the wavelet packet technique. *British Journal of Non-Destructive Testing,* 36(l):11, January 1994.
23. Gol'dshtein, R. V. and Mosolov, A. B. Fractal cracks. *Journal of Applied Mathematics and Mechanics,* 56(4):563, 1992.
24. Graham, M. D., Lane, S. L., and Luss, D. Proper orthogonal decomposition analysis of spatiotemporal temperature patterns. *Journal of Physical Chemistry,* 97(4):889, 1993.
25. Hingle, H. T. A practical method of machine tool condition monitoring of component surface finish data. *Proceedings of SPIE,* 803: 108, 1987.

26. Hwang, T. W. and Zhang, G. Analysis of elastoplastic deformation observed on machined surfaces. In K. F. Ehmann, editor, *Manufacturing Science and Engineering,* 64: 553, New Orleans, November 1993. ASME. Proceedings of 1993 ASME Winter Annual Meeting, PED.
27. SAS Institute Inc. *SAS/STAT User's Guide, Vol. 1 and 2, 4th Edition, Version 6.* Cary, NC, 1990.
28. Jones R. M. A guide to the interpretation of machinery vibration measurements, part 1. *Sound and Vibration,* 24, May 1994.
29. Kozek, W. Matched generalized labor expansion of nonstationary processes. *The Twenty Seventh Asilomar Conference on Signals, Systems, & Computers,* 1: 499, 1993.
30. Ladd, M. D. *Detection of Machinery Faults in Noise Using Wavelet Transform Techniques.* The University of Texas, Austin, 1995.
31. Ling, F. F. Scaling law for contoured length of engineering surfaces. *Journal of Applied Physics,* 62(6):2570, September 1987.
32. Ling, F. F. The possible role of fractal geometry in tribology. *Tribology Transactions,* 32(4):497, 1989.
33. Majumdar A. and Bhushan B. Fractal model of elastic-plastic contact between rough surfaces. *Trans ASME-Journal of Tribology,* 113(l):1, January 1991.
34. Malkin, S. *Grinding Technology.* Ellis Horwood Limited, Chichester, England, 1989.
35. Mallat, S. G. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 11(7):674, July 1989.
36. Mandelbrot, B. B. *The Fractal Geometry of Nature.* W. H. Freeman and Co., San Francisco, 2nd edition, 1983.
37. Marks, J. and Elbestawi, M. A. Development of machine tool condition monitoring system using pattern recognition. *ASME Winter Annual Meeting: Sensors and Controls for Manufacturing,* 33:89, 1988.
38. Martin, E. B., Morris, A. J., and Zhang, J. Process performance monitoring using multivariate statistical process control. *IEE Proceedings on Control Theory Applications,* 143:2, 1996.
39. McKoewn, P. A. The role of precision engineering in manufacturing of the future. *Annals of the CIRP,* 36:2:495, 1987.
40. Meguid, S. A. *Engineering Fracture Mechanics.* Elsevier Applied Science, London, 1989.
41. Monsen, P. T. and Manolakos, E. S. Helicopter gearbox fault detection and diagnosis using analog neural networks. *The twenty seventh Asilomar Conference on Signals, Systems, and Computers,* 1: 381, 1993.
42. Peitgen, H. O. and Saupe, D. *The Science of Fractal Images.* Springer-Verlag, New York, 1988.
43. Peters, J., Vanherck, P., and Sastrodinoto, M. Assessment of surface typology analysis techniques. *Annals of the CIRP,* 28:2:539, 1979.
44. Esterline Federal Products. Surfanalyzer 5000. User's Manual, 1993.
45. Chang, T. H., DeVor, R. E., and Sutherland, J. W. *Statistical Quality Design and Control.* Macmillan Publishing Co., New York, 1992.
46. Rao, P., Taylor, F., and Harrison, G. Real-time monitoring of vibration using the wigner distribution. *Sound and Vibration,* p. 22, May 1990.
47. Resnikoff, H. L. Wavelets and adaptive signal processing. *Optical Engineering,* 31(6):1229, June 1992.
48. Rioul, O. and Vetterli, M. Wavelets and signal processing. *IEEE Signal Processing Magazine,* p. 14, October 1991.
49. Rohrbaugh, R. A. Application of time-frequency analysis to machinery condition assessment. In *The twenty-seventh Asilomar Conference on Signals, Systems, & Computers,* 2:1455, 1993.
50. Schertzer, D. and Lovejoy, S. Nonlinear variability in geophysics: Multifractal simulations and analysis. In L. Pietronero, editor, *Fractals' Physical Origin and Properties,* p. 49. Plenum Press, New York, 1988.
51. Schroeder, M. *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise.* W. H. Freeman and Co., New York, 1991.
52. Sen, A. and Srivastava, M. *Regression Analysis: Theory, Methods, and Applications.* Springer-Verlag, New York, 1990.

53. Serridge, M. The use of cepstrum analysis in the vibration diagnostics of pulp and paper machinery problems. In *Proceedings of the 3rd International Conference on Condition Monitoring,* p. 47, 1991.
54. Sirovich, L. and Keefe, L. R. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America,* 4(3):519, March 1987.
55. Sottile, J. and Holloway, L. E. An overview of fault monitoring and diagnosis in mining equipment. *IEEE Transactions on Industry Applications,* 30(5):1326, September/October 1994.
56. Spiewak, S. A predictive monitoring and diagnosis system for manufacturing. *Annals of the CIRP: manufacturing technology,* 40(l):401, 1991.
57. Srinivasan, R. S. *A Theoretical Framework for Functional Form Tolerances in Design for Manufacturing.* The University of Texas, Austin, 1994.
58. Srinivasan, R. S. and Wood, K. L. A form tolerancing theory using fractals and wavelets. *Journal of Mechanical Design,* 119:2:185, June 1997.
59. Srinivasan, R. S., Wood, K.L., and McAdams, D. A methodology for functional tolerancing in design for manufacturing. *Research and Engineering Design,* 8:2:99, 1996.
60. Stanley, H. E. Fractals and multifractals: The interplay of physics and geometry. In A. Bunde, editor, *Fractals and Disordered Systems,* p. 1. Springer-Verlag, Berlin, 1991.
61. Strang, G. Wavelets and dilation equations: A brief introduction. *SIAM Review,* 31(4):614, December 1989.
62. Takayasu, H. A deterministic model for fracture. *Progress of Theoretical Physics,* 74(6):1343, December 1985.
63. Tonshoff, H. K., Wulfsberg, J. P., and Kals, H. J. J. Developments and trends in monitoring and control of machining processes. *Annals of the CIRP,* 37:2:611, 1988.
64. Traver, D. and Foley, D. Diagnosing gear and motor problems with signature analysis. *In Proceedings of the 11th Biennial conference on mechanical Vibration and Noise: Mechanical Signature Analysis,* 7: 39, 1987.
65. Trent, E. M. *Metal Cutting.* Butterworths, London, 1984.
66. Tsonis, A. A. A fractal study of dielectric breakdown in the atmosphere. In D. Schertzer, editor, *Nonlinear Variability in Geophysics: Scaling and Fractals,* p. 167. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1991.
67. Tumer, I. Y., Srinivasan, R. S., and Wood, K. L. Investigation of characteristic measures for the analysis and synthesis of precision-machined surfaces. *Journal of Manufacturing Systems,* 14(5):378, September/October 1995.
68. Tumer, I. Y., Thompson, D. C., Wood, K. L., and Crawford, R. H. Characterization of surface fault patterns with application to a layered manufacturing process. *Accepted for publication in the Journal of Manufacturing Systems,* April 1997.
69. Tumer, I. Y., Wood, K. L., and Busch-Vishniac, I. J. Improving manufacturing precision using the Karhunen-Loève transform. In *1997 ASME Design for Manufacturing Conference, Integrated Design Symposium,* volume DETC97-DFM4347 on CDROM, September 1997.
70. Turner, I. Y., Wood, K. L., and Busch-Vishniac, I. J. Monitoring fault condition during manufacturing using the Karhunen-Loève transform. In *1997 ASME Mechanical Vibration and Noise Conference, System Health Monitoring Symposium,* volume DETC97-VIB4234 on CDROM, September 1997.
71. Wardle, F. P. Vibration forces produced by the waviness of the rolling surfaces of thrust loaded ball bearings—parts I and 2. *Proceedings of the Institution of Mechanical Engineers,* 202(C5):305, 1988.
72. Whitehouse D. J. Typology of manufactured of surfaces. *Annals of the CIRP,* 3:417, 1971.
73. Whitehouse D. J. Surfaces - a link between manufacture and function. In *Proceedings of the Institution of Mechanical Engineers,* 192:179, 1978.
74. Whitehouse D. J. *Handbook of Surface Metrology.* Institute of Physics Publishing, Bristol, UK, 1994.
75. Whitehouse D. J. and Zheng K.G. The use of dual space-frequency functions in machine tool monitoring. *Measurement Science and Technology,* 3:796, 1992.

76. Wornell G. W. Synthesis, analysis, and processing of fractal signals. *Technical Report 566, Massachussets Institute of Technology,* Cambridge, October 1991.
77. Wu, J. M., Lee, J. Y., and Liou, C. Y. Diagnoses for machine vibrations based on self organization neural network. In *IECON'91,* P. 1506, 1991.
78. Wylie C. R. and Barrett L. C. *Advanced Engineering Mathematics.* McGraw-Hill, NewYork, 1982.
79. Ying S. P. and Dennison E. E. Application of coherence technique for noise in power plants. *Noise Control Engineering,* 15/2:81, September/October 1980.
80. Zahorian S. A. and Rothenberg M. Principal-components analysis for low-redundancy encoding of speech spectra. *Journal of the Acoustical Society of America,* 69:3:832, March 1981.
81. Zemel M. I. and Otto K.N. Use of injection molding simulation to assess critical dimensions and assign tolerances. In *The 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference,* volume DETC96-DFMI277 on CDROM, 1996.

2

Determination of Achievable Process Tolerances with Respect to Manufacturing Process Capability

Chiu-Chi Wei
Chung-Hua University

Chih-Hung Tsai
Ta-Hwa Institute of Technology

Chie-Bein Chen
Ming-Chuan University

2.1 Introduction
2.2 Traditional Manual Tolerance Charting
2.3 Computer-Aided Tolerance Charting
Matrix Approach
2.4 Mathematical Model
2.5 Computer-Aided Cost-Tolerance Optimization
Genetic Algorithm [27] • Simulated Annealing [31] • Taguchi Method [9]
2.6 Conclusion

Tolerance design plays a vital role in the manufacturing process planning. The tolerance should not be loosely allocated to degrade the product performance or strictly assigned to escalate the production cost. Traditional design of tolerance relies on personal experience and expertise; thus, the tolerance designed may be feasible but not economical. With the advent of the computer technology, many complicated algorithms for allocating the process tolerance have been developed, and searching for the optimal design of tolerance has become possible. This article intends to reveal the recent computational techniques of tolerance design to minimize the manufacturing cost and production loss.

2.1 Introduction

Tolerance design can be divided into three categories: (1) tolerance design of the final product specification to achieve required functionality; (2) tolerance design of the component specification to assure successful assembled product [1, 2, 3, 4, 5]; and (3) tolerance design of the manufacturing process to meet component specification [6, 7, 8, 9, 10]. Generally, the tolerance of the first two categories is called the product tolerance, while the third category is often named as process tolerance.

The use of the tolerance is inevitable due to the impossibility of producing two parts that have precisely the same dimensions. This could be attributed to the fact that many uncontrollable or random factors often exist during the production process. Tolerances specified on the blueprint by design engineers are specifications targeted to ensure a product's functional performance, and are often designed in such a way that the critical dimensions are specified with strict allowable tolerances, which then become the quality criteria of the final product inspection. Assigning tolerance to the nominal dimension of a component is one of the important factors to ensure that the product assembled will be within the functional requirements, and assembly tolerance stackup can be either worst-case or statistical. The worst-case tolerance stackup assumes that all components are produced within the tolerance limit; as a result, 100% inspection is needed to eliminate the possibility of using any nonconforming component in an assembly. Mathematically, the addition of the component tolerances shall be less than or equal to the assembly tolerance specification. On the other hand, the statistical tolerance stackup accepts slight defective assemblies; this matches the reality more closely. However, the assembly tolerance must be normally distributed and the component dimensions must be independent. Mathematically, the square root of the addition of the squared component tolerances shall be less than or equal to the standard deviation of the assembly tolerance. When the finalized blueprints of components are distributed for production from the research division, the process engineer designs a sequence of operations in order to produce the component as required. This is called process planning, which includes various tasks such as design of jig and fixture, tooling, NC program, standard operation sheet, etc. One of the important tasks in planning a process is the assignment of the tolerance to the working dimension of each operation. The process tolerance represents the control limits that should be strictly complied to by each operation to ensure that the final accumulated dimension of the component is within the blueprint specification. The amount of stock removal needed for each operation to reach the required precision, the machining accuracy of the equipment, the identification of the tolerance chain linking the consecutive reference and processed surfaces, and the tolerance stackup resulting from a series of operations must be considered while assigning process tolerances to various operations. The machining accuracy refers to the limitation or attainable tolerances of individual operation, and is frequently expressed in terms of the upper lower tolerance bounds. Specifically, these tolerance bounds represent the economical precision that can be held by a specific machine, and the engineers should design products with these tolerances in mind. Process tolerances are usually designed based on intuition and best guess of the engineers, thus being vulnerable to overestimation or underestimation. Loosely estimated tolerances may cause problems in the succeeding operations, while unnecessarily tight tolerances, on the other hand, escalate the manufacturing cost. One of the techniques used to widen the overestimated tolerances is called tolerance chart balancing, which can be performed either manually [11, 12, 13] or mathematically [2, 4, 14, 15, 16, 17, 18, 19, 22]. Among several other techniques, Nagarwala et al. [2] have proposed a slope-based method for solving the tolerance-process selection problem, wherein the cost function is approximated by a sequence of linear functions. Ngoi and Seng [4] designed tolerances by concurrently minimizing the nonlinear manufacturing cost and maximizing the working dimensions. Wei and Lee [10] have included the process capability index of the machinery in their tolerance design problem; therefore, the scrap rate of each operation and the entire process can be predicted with respect to the tolerances assigned. Ngoi [14] has proposed a linear programming model, coupled with identification schemes of tolerance chain, to determine the total additional tolerances to be added to the existing process tolerances. A weighting system is involved to evaluate the relative importance of operations. Unfortunately, this model does not consider the process capability of the equipment, thereby rendering the allocated tolerance to be uneconomic. Lee et al. [18] have used nonlinear design functions to determine the coefficients of the nonlinear cost equation. Lin et al. [19] have proposed a tolerance-process selection formulation using nonlinear objective function and constraints, wherein a genetic algorithm is applied to obtain the least-cost tolerances allocation. This article limits the discussion to process tolerance; therefore, the tolerance, hereafter, means process tolerance.

2.2 Traditional Manual Tolerance Charting

A traditional tolerance chart is a manual and graphical representation for planning the process tolerance stackup when the machining of component involves several independent tolerance chains. The process tolerance stackup can only be avoided by employing net shaped processes, and it is seldom avoided even when the manufacturing processes are continuously improved. When developing a manual tolerance chart, the following issues are decided in advance: the sequence of operations, the required machinery, and the proper tools. Moreover, some basic principles must be complied with:

1. The maximum possible tolerance should be allocated to each individual cut without violating the blueprint tolerance specification.
2. Tolerance allocated to each cut should be consistent with the corresponding machining accuracy.
3. Maximum and minimum stock removals for each cut should be feasible with respect to the machining equipment.

Figure 2.1 demonstrates an example adopted from Wade [13] describing the procedure of manually developing a traditional tolerance chart.

A line with an arrow at one end and dot at the other end represents a machining cut. A balance dimension is the algebraic combination of two or more cuts and has a dot at both ends. A dot denotes a locating or referenced surface and an arrow represents a newly machined surface. A B/P dimension is the heavy black line drawn between two machined surfaces. Every B/P value has a corresponding resultant dimension on the chart. The resultant dimension represents the actual value obtained after machining. It is noted that all B/P tolerances must be expressed in the equal bilateral tolerance system in the tolerance chart. The detailed steps of constructing the tolerance chart are briefly described below [20]:

Step 1: The outline of the component is drawn at the top of the chart. Feature plane lines and machining cut lines are introduced. All solid cuts are indicated in the stock removal column. Diameters are indicated against the outline of the component using balloons. The appropriate number $(N+1)$ of feature plane lines corresponding to the N blueprint dimensions are drawn vertically. The extra feature plane line corresponds to the surface which receives the initial facing cut and serves as a datum for subsequent cuts.

Step 2: Balance dimensions are installed between feature planes generated by machining cuts which contribute to the tolerance on one or more mean B/P values. In addition, B/P tolerances are budgeted among cuts present in all schematics, due consideration being given to (i) the accuracy of the machines performing each cut, and (ii) the occurrence of one or more machined cuts in the schematics for several B/P dimensions. The process engineer seeks to maximize the tolerance on cuts that occur in each B/P dimension's schematic since this minimizes total manufacturing cost.

Step 3: The tolerances on cuts from the previous step are entered in the chart. Other entries in this step are (i) mean B/P values, (ii) their tolerances, and (iii) line numbers involved for all stock removals on secondary cuts and balance dimensions.

Step 4: Schematics for B/P dimensions do not include all the machining cuts in the process plan. Tolerances on cuts not appearing in the schematics do not influence the B/P resultant dimensions but contribute to stackup on stock removal tolerances. An additional set of schematics is constructed for secondary cuts on feature planes where the stackup of stock removal tolerance must be controlled.

Step 5: Mean stock removals are computed by the careful consideration of the relevant machine/material system for each constituent cut in the schematic. Due consideration must again be given to the capability of the process assigned to perform the stock removal cut.

Step 6: Mean dimensions are computed for machined lengths and balance dimensions, starting from the bottom of the chart and working towards the top. Equations involving machining dimensions and balance dimensions are solved through inspection and backtracking until all unknown mean dimensions are obtained.

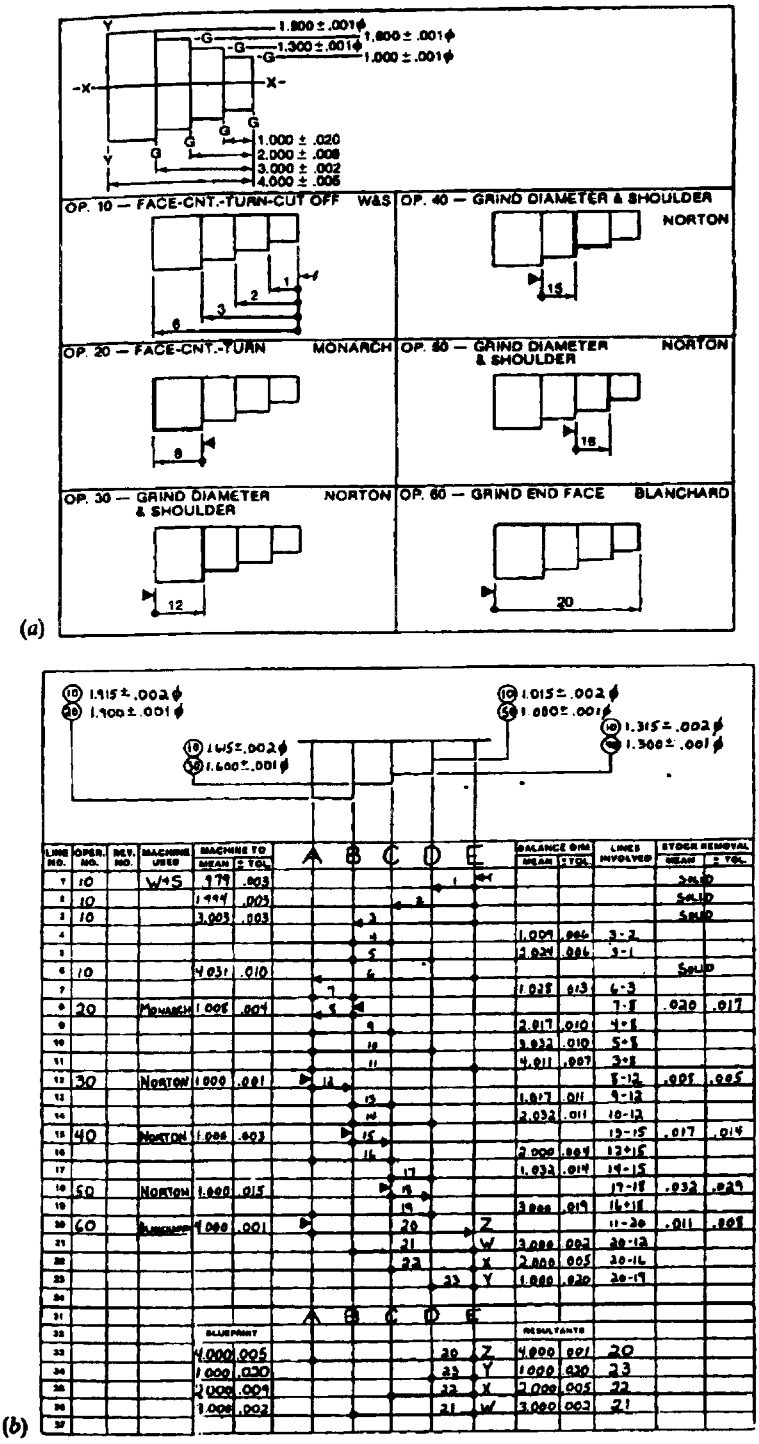


FIGURE 2.1 Traditional manual tolerance charting.

2.3 Computer-Aided Tolerance Charting

Tolerance chart technique was proposed in the 1950s [11, 12, 13], and has been widely applied to the component manufacturing for assigning the process tolerance. Along with the advent of computer technology in 1960s, tolerance charting has undergone numerous improvements and early works focus mainly on the direct translation of the manual procedures into computer steps. Although tedious computations were replaced by the computer, human intervention was still required at certain decision points. Recent developments adopted more scientific approaches, which has led to many successful packages. In those systems the requirements and constraints, such as machining accuracy and tolerance stackup, were included to form a mathematical model, which was then solved using linear programming, non-linear programming, or other heuristic algorithms.

Matrix Approach

Ngoi and Kai [21] have developed a matrix approach to tolerance charting without the need of the representation of the machining sequence. The dimensions and tolerances between surfaces are organized into an input matrix, which then undergoes a series of operations to yield the final matrix that contains the resultant dimensions and tolerances between any two surfaces of the component. The matrix approach is applied to the component shown in Fig. 2.2 adopted from [21]. The component is to be machined through ten operations including turning, milling, carburising, and grinding. The tolerance chart for the component is prepared in Table 2.1, which indicates the locating or reference surfaces and the processed surfaces. All capital letters represent the workpiece surfaces: the surface BC and GC are used to indicate the carburised surface of B and G, whereas B1 and G1 are the newly manufactured surface of B and G.

For ease of representation, instead of using the real values of the dimension and tolerance, the element of the matrix will be symbolically expressed. For example, GA represents the dimension between face G and face A in the dimension matrix. Similarly, GA denotes the tolerance between face G and face A in the tolerance matrix. The detailed procedures are described below:

Step 1. Input matrix The processed surface, reference surface, and working dimensions are organized into a matrix shown in Fig. 2.3. The input matrix may represent either the input dimension matrix or input tolerance matrix, depending on whether the cells represent dimensions or tolerances. From the input matrices, a series of operations on the matrix is conducted to derive the dimension and tolerance between other surfaces.

Step 2. Operation 1 The matrix is checked row by row from bottom to top. If a row has more than one dimension/tolerance element, then the leftmost element is the reference element and the other elements of the same row are parent elements. The parent elements are taken one at a time, from

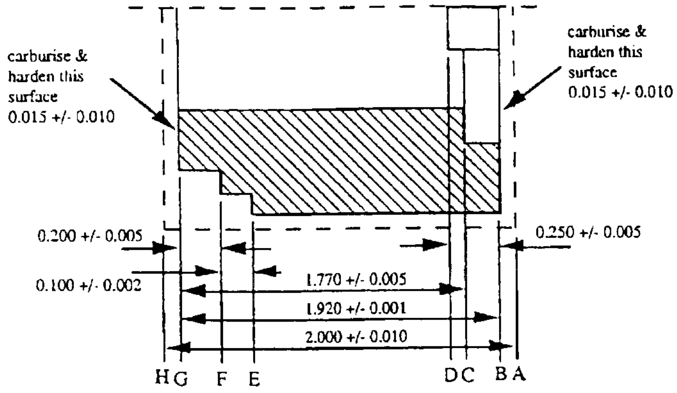


FIGURE 2.2 The component.

TABLE 2.1 The Tolerance Chart

Operation No.	Reference Surface	Processed Surface	Machine	Working Dimension	Working Tolerance +/−	Stock Removal	Total Tolerance
10	A	Surface G	Turret lathe	1.970	0.002	0.030	0.012
20a	G	Surface B	Turret lathe	1.940	0.002	0.030	0.004
20b	G	Surface C	Turret lathe	1.780	0.002	0.160	0.004
30a	B	Surface F	Turret lathe	1.730	0.002		
30b	F	Surface E	Turret lathe	0.100	0.002		
40	B	Surface D	Simplex mill	0.260	0.002	0.260	0.002
50a	G	Surface GC	Carb. & H.T. Equipment	0.025	0.005		
50b	B	Surface BC	Carb. & H.T. Equipment	0.025	0.005		
60	B	Surface G1	Rotary surface grinder	1.930	0.0005	0.010	0.0025
70	G1	Surface B1	Rotary surface grinder	1.920	0.0005	0.010	0.001

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	GA	0	0	0	0	0	0	0	0	0
G	GA	0	BG	CG	0	0	0	GcG	0	0	0
B	0	BG	0	0	FB	0	DB	0	BcB	G1B	0
C	0	CG	0	0	0	0	0	0	0	0	0
F	0	0	FB	0	0	EF	0	0	0	0	0
E	0	0	0	0	EF	0	0	0	0	0	0
D	0	0	DB	0	0	0	0	0	0	0	0
Gc	0	GcG	0	0	0	0	0	0	0	0	0
Bc	0	0	BcB	0	0	0	0	0	0	0	0
G1	0	0	G1B	0	0	0	0	0	0	0	B1G1
B1	0	0	0	0	0	0	0	0	0	B1G1	0

FIGURE 2.3 The input dimension-tolerance matrix.

left to right. Depending on the type of matrix, the difference between or sum of the parent element and the reference element is then calculated. The calculated value is entered into a cell, and the location of the cell is defined as follows:

$$\begin{aligned}\text{Column of cell} &= \text{Column of the cell of its parent element}\\ \text{Row of the cell} &= \text{Column of the cell of the reference element}\end{aligned}$$

The operation is completed when the top row is reached (Fig. 2.4); the general expression is:

$$YZ = XY - ZX \text{ (dimension matrix)}$$

For example:

$$\begin{aligned}AB1 &= GA - B1G\\ GF &= BG - FB\end{aligned}$$

$$YZ = XY + ZX \text{ (tolerance matrix)}$$

For example:

$$\begin{aligned}AB1 &= GA + B1G\\ GF &= BG + FB\end{aligned}$$

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	GA	GA-BG	GA-CG	GA-FG	GA-EG	GA-DG	GA-GcG	GA-BcG	GA-G1G	GA-B1G
G	GA	0	BG	CG	BG-FB	BG-EB	BG-DB	GcG	BG-BcB	BG-G1B	BG-B1B
B	0	BG	0	0	FB	FB-EF	DB	0	BcB	G1B	G1B-B1G1
C	0	CG	0	0	0	0	0	0	0	0	0
F	0	0	FB	0	0	EF	0	0	0	0	0
E	0	0	0	0	EF	0	0	0	0	0	0
D	0	0	DB	0	0	0	0	0	0	0	0
Gc	0	GcG	0	0	0	0	0	0	0	0	0
Bc	0	0	BcB	0	0	0	0	0	0	0	0
G1	0	0	G1B	0	0	0	0	0	0	0	B1G1
B1	0	0	0	0	0	0	0	0	0	B1G1	0

(a) Dimension Matrix

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	GA	GA+BG	GA+CG	GA+FG	GA+EG	GA+DG	GA+GcG	GA+BcG	GA+G1G	GA+B1G
G	GA	0	BG	CG	BG+FB	BG+EB	BG+DB	GcG	BG+BcB	BG+G1B	BG+B1B
B	0	BG	0	0	FB	FB+EF	DB	0	BcB	G1B	G1B+B1G1
C	0	CG	0	0	0	0	0	0	0	0	0
F	0	0	FB	0	0	EF	0	0	0	0	0
E	0	0	0	0	EF	0	0	0	0	0	0
D	0	0	DB	0	0	0	0	0	0	0	0
Gc	0	GcG	0	0	0	0	0	0	0	0	0
Bc	0	0	BcB	0	0	0	0	0	0	0	0
G1	0	0	G1B	0	0	0	0	0	0	0	B1G1
B1	0	0	0	0	0	0	0	0	0	B1G1	0

(b) Tolerance Matrix

FIGURE 2.4 The dimension and tolerance matrix after operation 1.

Step 3. Operation 2 The upper half of the matrix divided by the diagonal is inspected cell by cell from top to bottom. The aim is to determine the location of the reference cell and the parent cell. The value of the empty cell of a dimension or tolerance matrix is the absolute difference between or sum of the value of the reference and the value of the parent cell. If a cell is empty then the leftmost element of that row is the reference element, the column face of the reference element and the column face of the empty cell are noted. The parent cell is located as follows:

$$\begin{aligned}\text{Column of parent cell} &= \text{Column of the cell of the empty element}\\ \text{Row of the parent cell} &= \text{Column of the cell of the reference element}\end{aligned}$$

Using the reference surfaces, the dimensions and tolerances are derived as follows:

$$Bc = MOD\ (BG - CG)$$

where BG is the reference surface in the dimension matrix,

$$Bc = BG + CG$$

where BG is the reference surface in the tolerance matrix.

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	GA	BA	CA	FA	EA	DA	GcA	BcA	G1A	B1A
G	GA	0	BG	CG	FG	EG	DG	GcG	BcG	G1G	B1G
B	0	BG	0	BG-CG	FB	EB	DB	BG-GcG	BcB	G1B	B1B
C	0	CG	0	0	CG-FG	CG-EG	CG-DG	CG-GcG	CG-BcG	CG-G1G	CG-B1G
F	0	0	FB	0	0	EF	FB-DB	FB-GcB	FB-BcB	FB-G1B	FB-B1B
E	0	0	0	0	EF	0	EF-DF	EF-GcF	EF-BcB	EF-G1F	EF-B1F
D	0	0	DB	0	0	0	0	DB-GcB	DB-BcB	DB-G1B	DB-B1B
Gc	0	GcG	0	0	0	0	0	0	GcG-BcG	GcG-G1G	GcG-B1G
Bc	0	0	BcB	0	0	0	0	0	0	BcB-G1B	BcB-B1B
G1	0	0	G1B	0	0	0	0	0	0	0	B1G1
B1	0	0	0	0	0	0	0	0	0	B1G1	0

(a) Dimension Matrix

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	GA	BA	CA	FA	EA	DA	GcA	BcA	G1A	B1A
G	GA	0	BG	CG	FG	EG	DG	GcG	BcG	G1G	B1G
B	0	BG	0	BG+CG	FB	EB	DB	BG+GcG	BcB	G1B	B1B
C	0	CG	0	0	CG+FG	CG+EG	CG+DG	CG+GcG	CG+BcG	CG+G1G	CG+B1G
F	0	0	FB	0	0	EF	FB+DB	FB+GcB	FB+BcB	FB+G1B	FB+B1B
E	0	0	0	0	EF	0	EF+DF	EF+GcF	EF+BcB	EF+G1F	EF+B1F
D	0	0	DB	0	0	0	0	DB+GcB	DB+BcB	DB+G1B	DB+B1B
Gc	0	GcG	0	0	0	0	0	0	GcG+BcG	GcG+G1G	GcG+B1G
Bc	0	0	BcB	0	0	0	0	0	0	BcB+G1B	BcB+B1B
G1	0	0	G1B	0	0	0	0	0	0	0	B1G1
B1	0	0	0	0	0	0	0	0	0	B1G1	0

(b) Tolerance Matrix

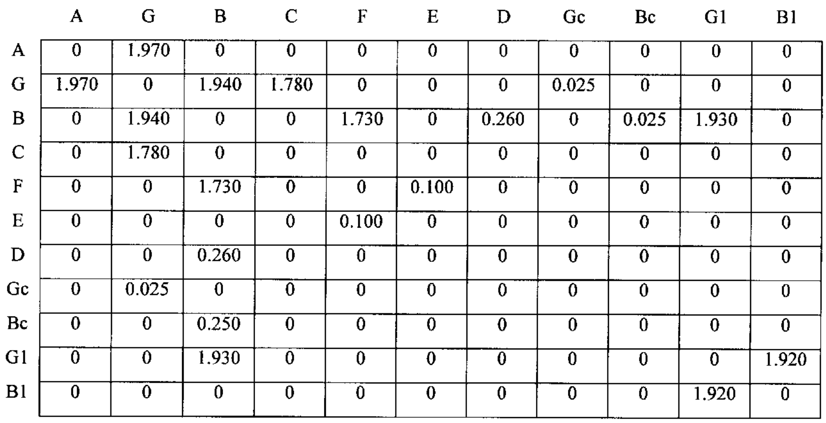
FIGURE 2.5 The dimension and tolerance matrix after operation 2.

The operation is completed when the value of every cell in the upper half of the diagonal is determined (Fig. 2.5).

The method described is now applied to the tolerance chart in Table 2.1. The input dimension and tolerance matrix are generated in Fig. 2.6; the matrices after operation 1 and operation 2 are shown in Fig. 2.7 and Fig. 2.8. Table 2.2 is the resultant dimension and tolerance. It can be seen that all resultant dimensions are within those specified on the blueprint. This matrix technique is efficient and easy to implement on computer.

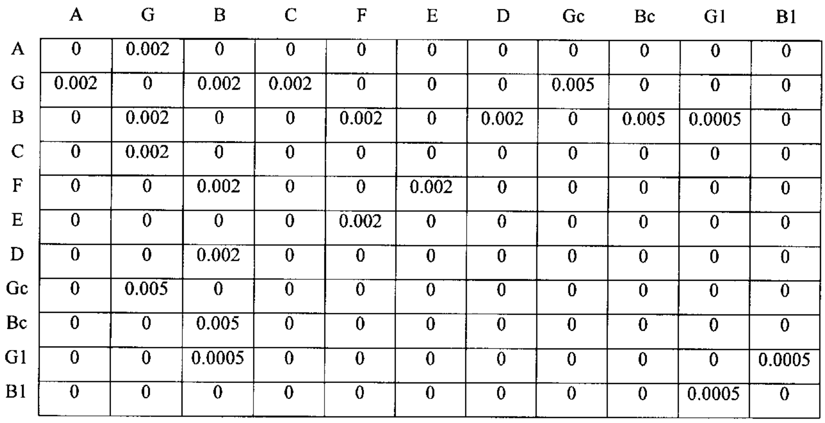
2.4 Mathematical Model

It is widely known that the stricter the process tolerance, the higher the manufacturing cost in terms of the precision of the machinery and tools, the required machining time, and the special environmental control, such as humidity, temperature, and cleanness of the working area. Therefore, it has become the general objective that the process tolerance should be designed as large as possible to reduce the production



	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	1.970	0	0	0	0	0	0	0	0	0
G	1.970	0	1.940	1.780	0	0	0	0.025	0	0	0
B	0	1.940	0	0	1.730	0	0.260	0	0.025	1.930	0
C	0	1.780	0	0	0	0	0	0	0	0	0
F	0	0	1.730	0	0	0.100	0	0	0	0	0
E	0	0	0	0	0.100	0	0	0	0	0	0
D	0	0	0.260	0	0	0	0	0	0	0	0
Gc	0	0.025	0	0	0	0	0	0	0	0	0
Bc	0	0	0.250	0	0	0	0	0	0	0	0
G1	0	0	1.930	0	0	0	0	0	0	0	1.920
B1	0	0	0	0	0	0	0	0	0	1.920	0

(a) Dimension Matrix



	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	0.002	0	0	0	0	0	0	0	0	0
G	0.002	0	0.002	0.002	0	0	0	0.005	0	0	0
B	0	0.002	0	0	0.002	0	0.002	0	0.005	0.0005	0
C	0	0.002	0	0	0	0	0	0	0	0	0
F	0	0	0.002	0	0	0.002	0	0	0	0	0
E	0	0	0	0	0.002	0	0	0	0	0	0
D	0	0	0.002	0	0	0	0	0	0	0	0
Gc	0	0.005	0	0	0	0	0	0	0	0	0
Bc	0	0	0.005	0	0	0	0	0	0	0	0
G1	0	0	0.0005	0	0	0	0	0	0	0	0.0005
B1	0	0	0	0	0	0	0	0	0	0.0005	0

(b) Tolerance Matrix

FIGURE 2.6 Input dimension and tolerance matrix.

cost, provided that the functional performance of the component will not be sacrificed. Figure 2.9 demonstrates the relationship between tolerance and cost [1].

Various cost-tolerance models have been proposed to approximate the relationship between tolerance and cost, including the reciprocal squared model, the exponential model, the Michael-Siddall squared model, and the discrete model [22]. Dong and Hu [23] tested different curves against empirical data and concluded that the polynomial models are found to give more accurate results than others. However, Ostwald and Blake [24] indicated that Bennet and Gupta's equation performed best, while Speckhart's was found to yield the lowest percent maximum error. The cost-tolerance models proposed are summarized as follows:

(i) Power model (Bennet and Gupta, 1969):

$$C(\delta) = a_0\delta^{a_1}$$

(ii) Exponential model (Speckhart, 1972):

$$C(\delta) = a_0 e^{-\delta_1}$$

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	1.970	0.030	0.190	1.760	1.660	0.290	1.945	0.055	1.960	0.040
G	1.970	0	1.940	1.780	0.210	0.310	1.680	0.025	1.915	0.010	1.930
B	0	1.940	0	0	1.730	1.630	0.260	0	0.025	1.930	0.010
C	0	1.780	0	0	0	0	0	0	0	0	0
F	0	0	1.730	0	0	0.100	0	0	0	0	0
E	0	0	0	0	0.100	0	0	0	0	0	0
D	0	0	0.260	0	0	0	0	0	0	0	0
Gc	0	0.025	0	0	0	0	0	0	0	0	0
Bc	0	0	0.250	0	0	0	0	0	0	0	0
G1	0	0	1.930	0	0	0	0	0	0	0	1.920
B1	0	0	0	0	0	0	0	0	0	1.920	0

(a) Dimension Matrix

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	0.002	0.004	0.004	0.006	0.008	0.006	0.007	0.009	0.0045	0.005
G	0.002	0	0.002	0.002	0.004	0.006	0.004	0.005	0.007	0.0025	0.003
B	0	0.002	0	0	0.002	0.004	0.002	0	0.005	0.0005	0.001
C	0	0.002	0	0	0	0	0	0	0	0	0
F	0	0	0.002	0	0	0.002	0	0	0	0	0
E	0	0	0	0	0.002	0	0	0	0	0	0
D	0	0	0.002	0	0	0	0	0	0	0	0
Gc	0	0.005	0	0	0	0	0	0	0	0	0
Bc	0	0	0.005	0	0	0	0	0	0	0	0
G1	0	0	0.0005	0	0	0	0	0	0	0	0.0005
B1	0	0	0	0	0	0	0	0	0	0.0005	0

(b) Tolerance Matrix

FIGURE 2.7 Dimension and tolerance matrix (operation 1).

(iii) Reciprocal squared model (Spotts, 1973):

$$C(\delta) = \frac{a_0}{\delta^2}$$

(iv) Reciprocal powers model (Sutherland and Roth, 1975):

$$C(\delta) = a_0\delta^{-a_i}$$

(v) Reciprocal power and exponential hybrid model (Michael and Siddall, 1981):

$$C(\delta) = a_0\delta^{-a_1}e^{-a_2\delta}$$

TABLE 2.2 The Resultant Dimension and Tolerance

Component Surface		Resultant Dimension	Blueprint Dimension
G1	F	0.200 ± 0.0025	0.002 ± 0.005
C	G1	1.770 ± 0.0045	1.770 ± 0.005
D	B1	0.250 ± 0.003	0.250 ± 0.005
B1	Bc	0.015 ± 0.006	0.015 ± 0.010
G1	Gc	0.015 ± 0.0075	0.015 ± 0.010
E	F	0.100 ± 0.002	0.100 ± 0.002
G1	B1	1.920 ± 0.0005	1.920 ± 0.001

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	1.970	0.030	0.190	1.760	1.660	0.290	1.945	0.055	1.960	0.040
G	1.970	0	1.940	1.780	0.210	0.310	1.680	0.025	1.915	0.010	1.930
B	0	1.940	0	0.160	1.730	1.630	0.260	1.915	0.025	1.930	0.010
C	0	1.780	0	0	1.570	1.470	0.100	1.755	0.135	1.770	0.150
F	0	0	1.730	0	0	0.100	1.470	0.185	1.705	0.200	1.720
E	0	0	0	0	0.100	0	1.370	0.085	1.605	0.100	1.620
D	0	0	0.260	0	0	0	0	1.655	0.235	1.670	0.250
Gc	0	0.025	0	0	0	0	0	0	1.890	0.015	1.905
Bc	0	0	0.250	0	0	0	0	0	0	1.905	0.015
G1	0	0	1.930	0	0	0	0	0	0	0	1.920
B1	0	0	0	0	0	0	0	0	0	1.920	0

(a) Dimension Matrix

	A	G	B	C	F	E	D	Gc	Bc	G1	B1
A	0	0.002	0.004	0.004	0.006	0.008	0.006	0.007	0.009	0.0045	0.005
G	0.002	0	0.002	0.002	0.004	0.006	0.004	0.005	0.007	0.0025	0.003
B	0	0.002	0	0.004	0.002	0.004	0.002	0.007	0.005	0.0005	0.001
C	0	0.002	0	0	0.006	0.008	0.006	0.007	0.009	0.0045	0.005
F	0	0	0.002	0	0	0.002	0.004	0.009	0.007	0.0025	0.003
E	0	0	0	0	0.002	0	0.006	0.11	0.009	0.0045	0.005
D	0	0	0.002	0	0	0	0	0.009	0.007	0.0025	0.003
Gc	0	0.005	0	0	0	0	0	0	0.012	0.0075	0.008
Bc	0	0	0.005	0	0	0	0	0	0	0.0055	0.006
G1	0	0	0.0005	0	0	0	0	0	0	0	0.0005
B1	0	0	0	0	0	0	0	0	0	0.0005	0

(b) Tolerance Matrix

FIGURE 2.8 Dimension and tolerance matrix (operation 2).

(vi) Reciprocal model (Chase and Greenwood, 1990):

$$C(\delta) = \frac{a_0}{\delta}$$

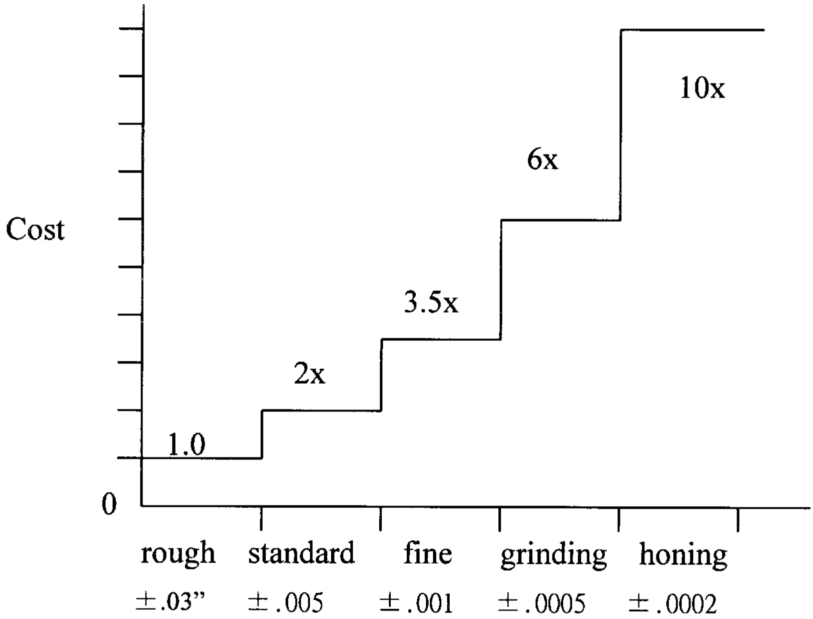


FIGURE 2.9 The relationship between tolerance and cost.

(vii) Reciprocal squared and exponential model (Sfantsikopoulos, 1990):

$$C(\delta) = a_0 + a_1 \frac{i(D)^{r+1}}{\delta^r}$$

where $i(D)$ is the ISO standard tolerance factors

(viii) Modified exponential model (Dong and Soom, 1990):

$$C(\delta) = a_o e^{-a_1(\delta - a_2)} + a_3$$

(ix) Combined reciprocal power and exponential model (Lee and Woo, 1991):

$$C(\delta) = a_0 + a_1 \delta^{-a_2} + a_3 e^{-a_4 \delta}$$

$$C(\delta) = a_0 + a_1 \delta + a_2 e^{-a_3 \delta}$$

(x) Combined linear and exponential model (Dong and Hu, 1991):

(xi) Cubic polynomial model (Dong and Hu, 1991):

$$C(\delta) = a_0 + a_1 \delta + a_2 \delta^2 + a_3^{12} \delta^3$$

where a_i is the constant, δ is the tolerance.

In summary, the manufacturing cost depends heavily on the machinery/tool/ material system. Different models may need to be developed for each operation conducted on a specific machine. Moreover, not all factors that contribute to the production cost were considered in the above models, such as material cost, overhead cost, tooling cost, rework cost, scrap cost, and quality loss. Thus, some researchers have proposed different objective functions that include different cost items. The objective of the mathematical models presented previously can be summarized as the following categories:

(i) Minimization of cost-tolerance function (Balakrishnan, 1993)

$$\text{Min} \sum C(\delta_j)$$

S.T. tolerance stackup constraints
machining accuracy constraints

(ii) Minimization of cost-tolerance function and rework cost (Lee et al., 1993)

$$\text{Min} \sum C(\delta_j) + Cr$$

S.T. tolerance stackup constraints
machining accuracy constraints

where Cr is the rework cost.

(iii) Minimization of machining, material and rework costs (Malek & Asadathorn, 1994)

$$\text{Min} \sum C(\delta_j) + M + R_1(p_1) + R_2(p_2)$$

S.T. tolerance stackup constraints
machining accuracy constraints

where R_1 and R_2 are the rework cost for under-sized and over-sized parts, p_1 and p_2 are the probability of reworking, M is the material cost.

(iv) Minimization of cost-tolerance function and quality loss (Cheng & Maghsoodloo, 1995)

$$\text{Min} \sum C(\delta_j) + E\{k_i(y_i - m_i)^2\}$$

S.T. tolerance stackup constraints
machining accuracy constraints

where E denotes expected quality loss,
k_i is constant,
y_i is a measurement of quality characteristic,
m_i is the target of y_i

(v) Minimization of scrap cost (Wei, 1997)

$$\text{Min} \sum C(\delta_i)\{r_i\}$$

S.T. tolerance stackup constraints
machining accuracy constraints

where r_i denotes the scrap rate of operation.

2.5 Computer-Aided Cost-Tolerance Optimization

Computer technology has been widely applied to tolerance design during the last two decades. Along with the concept of computer-aided design and computer-aided manufacturing, the tasks of tolerance design are combined with the CAD/CAM system. Farmer [25] has developed an interactive tolerance design module on a CAD system. Juster [26] has established experimental software named GEOTOL, which

TABLE 2.3 Summary of the Tolerance Design Expert System

Expert System	Data Input	Dimension	Tolerance	Method	Notes
Proplan (1984)	CAD database	2D	No	Graphic search	Inefficient for complex parts
Excap (1984)	Decision tree	2D	No	Backward chaining	No interface with CAD
Sipp (1985)	N/A	3D	Yes	Least-cost search	Tolerance for process selection
Tipps (1985)	CAD database	3D	Yes	Surface extraction	Interactive tolerance not detected
Turbo CAPP (1987)	CAD database	2D	Yes	Surface & bound extraction	Improper tolerances not detected
Xplan-R (1988)	N/A	2D	Yes	Hybrid	No interface with CAD
TVCAPP (1993)	CAD interface	3D	Yes	Interactive CAD	Tolerance verified vs. CSA standard

From Malck, L.A. and Asadathorn, N., Process and desigh tolerance in relation to manufacturing cost, *The Engineering Economist*, 40(1), 1994. Reprinted with the permission of the Institute of Industrial Engineers, 25 Technology Park, Norcross, GA 30092, 770-449-0461. Copyright©1994.

can perform a complete tolerance analysis without the designer's guidance. The technique of artificial intelligence has also been utilized in tolerance allocation. An AI system usually establishes a knowledge base defining the rules that are used to allocate the tolerance of component. Table 2.3 summarizes the tolerance design expert systems using the AI technique [1].

Abdou and Cheng [26] indicated that few systems have developed the important linkage between CAD and CAPP. Among the AI systems in Table 2.3, TVCAPP can verify and generate both dimensional and geometric tolerances, and produce alternatives for process plans with corresponding costs.

Most of the mathematical models described above are nonlinear programming problems, which are frequently solved using either nonlinear optimization packages or heuristic algorithms. The heuristic algorithms proposed include genetic algorithm, simulated annealing, and Taguchi method.

Genetic Algorithm [27]

GA uses three operators, the reproduction, the crossover, and the mutation. The reproduction operator allows the highly productive chromosomes (strings) to live and produce offspring in the next generation. The crossover operator, used with a specified probability, exchanges genetic information by splitting two chromosomes at a random site and joining the first part of one chromosome with the second part of another chromosome. Mutation introduces occasional changes of a random string position with a specified mutation probability. The general procedure for genetic algorithms in tolerance optimization problems is described as follows:

1. An appropriate chromosome representation should be defined to represent the combinations of design variables that correspond to the fitness or objective function values. The representation should be a one-to-one mapping in order to have a normal coding and decoding process.
2. The probabilities of crossover and mutation are specified. The population size and maximum number of generations are selected, and an initial population in the genetic system is generated.
3. Evaluate the objective function value or fitness of each solution in the current generation.
4. Apply the three operators to those solutions in the current generation to produce a new population.
5. Repeat steps 3 and 4 until the maximum number of generations is reached.

Simulated Annealing [31]

Simulated annealing was proposed in early 1980s. It is a simulation of the physical annealing process of metal heat treatment. It differs from iterative improvement algorithms in the sense that the latter accepts only those moves that will lead to the improvement of the results. The inherent problems with this kind of algorithm are that: (1) it can be easily trapped in a local optimum; and (2) it depends heavily on the initial solution. By allowing perturbations to move to a worse solution in a controlled fashion, the simulated annealing algorithm is able to jump out of a local optimum and potentially find a more promising downhill path. Moreover, the uphill moves (the acceptance of worse solutions) are carefully controlled. Although the global solution is not guaranteed, it consistently provides solutions close to the optimal one. Besides, with the capability of accepting uphill moves, this algorithm appears to be independent of the initial solution. The basic elements of the simulated annealing algorithm are listed below:

1. Configuration: a solution of the problem.
2. Move: a transition from one configuration to another.
3. Neighboring configuration: a result of a move.
4. Objective function: a measure of how good a solution is.
5. Cooling scheduling: how high the starting temperature should be, and the rules to determine (a) when the current temperature should be lowered, (b) by how much the temperature should be lowered, and (c) when the annealing process should be terminated.

The algorithm of the simulated annealing is described as the following pseudo-code:

```
Begin with an initial solution $s$ and temperature $t$
Determine the initial direction and step size of the move
Repeat
  Perform the following loop $M$ times
    Pick a neighbouring configuration $s'$ of $s$ by a random move
    Let $\Delta = ((f(s') - f(s))/f(s)$
    If $\Delta \leqq 0$ (downhill move)
      Set $s' = s'$
  Else (uphill move)
    Set $s' = s'$ with the probability $e^{-\Delta/t}$
  Update the direction and step size of moves based on the ratio of accepted to rejected moves
Lower $t$ until one of the terminating conditions is true
Return $s$
```

Taguchi Method [9]

The Taguchi method of design of experiments begins with identifying a response and classifying the product or process parameters as (a) control (or design) parameters, or (b) noise (or uncertain) parameters. Control parameters are the product characteristics whose nominal settings can be specified by the product designer. Noise parameters are the variables that cause performance variation during the lifespan of the product. The second step of the Taguchi method is to arrange these control and noise parameters in the orthogonal array developed by Taguchi, and then conduct experiment and collect data. Finally, the signal-to-noise ratios, as opposed to the ANOVA in the standard design of experiment approach, are used to analyze the experiments. The general procedure for applying the Taguchi method to tolerance design is as follows:

1. Consider the nominal values of dimensions with tolerances as control parameters, their corresponding tolerances as noise parameters, and the direct manufacturing costs as response.
2. Select a proper orthogonal array.
3. Incorporate the “constraint” column in the standard orthogonal array to represent the tolerance stackup constraint.
4. Calculate the tolerance stackup and response for each set (row) of parameters.
5. Select the row with the smallest response value and for which all the tolerance stackup constraints are satisfied as the best tolerance.

TABLE 2.4 The Component for Taguchi Method

Experiment Number	Control Factors			Noise Factors Level (Tolerance/Cost)			Constraint T	Response y
	A	B	C	Ta	Tb	Tc		
1	80	60	60	0(2/92)	0(2/92)	0(4/65)	8	249
2	80	60	60	1(4/65)	1(3/75)	1(6/42)	13	182
3	80	60	60	2(5/57)	2(5/57)	2(10/20)	20	134
4	80	60	60	0(2/92)	1(3/75)	2(10/20)	15	321
5	80	60	60	1(4/65)	2(5/57)	0(4/65)	13	187
6	80	60	60	2(5/57)	0(2/92)	1(6/42)	13	191
7	80	60	60	0(2/92)	2(5/57)	1(6/42)	12	191
8	80	60	60	1(4/65)	0(2/92)	2(10/20)	16[a]	177[a]
9	80	60	60	2(5/57)	1(3/75)	0(4/65)	12	197

[a] Optimal tolerance design.

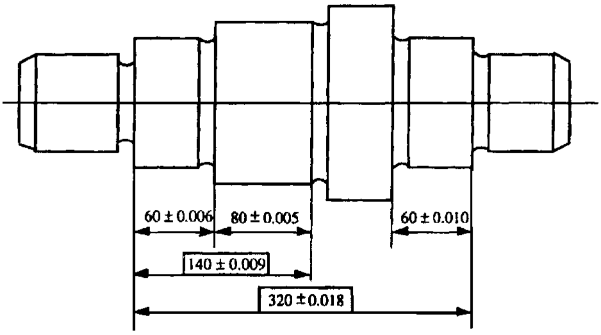


FIGURE 2.10 The component for Taguchi method.

The component shown in Fig. 2.10 is used to illustrate the Taguchi method, and Table 2.4 is the design of experiment indicating the optimal tolerance design. The three dimensions A, B, and C are the control factors, and the three tolerances are the noise factors. An L_9 (3^4) layout of the design of experiment is used. The minimal value of the manufacturing cost is $y_{\min} = 177$ with the tolerance stackup constraint $T \leqq 16$ being satisfied.

2.6 Conclusion

The purpose of the process planning is to design a sequence of operations to machine the component in such a way that the final product should be within the functional requirements specified on the blueprint. The tasks of process planning include selection of the machinery, design of the jig and fixture, preparation of the standard operation sheet, coding of the NC program and the like. To choose the suitable machine implies that the process planner must be aware of the machining accuracy of the equipment. The standard operation sheet specifies various items compiled by the operators, such as the required jig and fixture, NC program, inspection apparatus and the quality standard. The quality standard is mostly expressed as the dimensional tolerance to be maintained at each operation. This tolerance is usually designed by the process planner based on their intuition and best guess; as a result, the tolerance designed may not be economically feasible. To improve the quality of the tolerance design, researchers have proposed many mathematical models, both linear and nonlinear programming, coupled with the use of the computer in expectation of assigning the process tolerance to minimize the manufacturing cost, rework cost, scrap cost, and production loss. The optimization techniques presented include the genetic algorithm, simulated annealing, Taguchi method, and other searching algorithm. Moreover, many workers have explored the possibility of combining the tolerance design function with the process selection scheme. However, no existing system provides the capability that allows the users to interactively plan the process with the concurrent consideration of tolerance design to minimize the production cost. In other words, efforts still need to be devoted to transfer the experimental type system into a practical one. In addition, review of the literature indicates that a complete system integrating the CAD, CAPP (Computer-Aided Process Planning), and CAM is highly needed to improve the efficiency and quality of the product design and manufacturing cycle.

References

1. Malek, L. A. and Asadathorn, N., Process and Design Tolerance in Relation to Manufacturing Cost: A Review and Extension, *The Engineering Economist,* 40: 1: 73, 1994.
2. Nagarwala, M. Y., Pulat, P. S., and Raman, S., A Slope-Based Method for Least Cost Tolerance Allocation, *Concurrent Engineering,* 3: 4: 319, 1995.
3. Cheng, B. W. and Maghsoodloo, S., Optimization of Mechanical Assembly Tolerances by Incorporating Taguchi's Quality Loss Function, *Journal of Manufacturing Systems,* 14: 4: 264, 1995.
4. Ngoi, B. K. A. and Seng, M. S. M., Tolerance Synthesis Adopting a Nonlinear Programming Approach, *International Journal of Advanced Manufacturing Technology,* 11: 387, 1996.
5. Wei, C., Designing Process Tolerances to Minimize Loss of Nonconforming Parts, *IEEE Transactions on Components, Packaging and Manufacturing Technology, Part C: Manufacturing,* 20: 4: 1, 1998.
6. Ngoi, B. K. A. and Fang, S. L., Computer Aided Tolerance Charting, International *Journal of Production Research,* 32: 8: 1939, 1993.
7. Zhang, C. and Wang, H. P., Optimal Process Sequence Selection and Manufacturing Tolerance Allocation, *Journal of Design and Manufacturing,* 3: 135, 1993.
8. Ngoi, B. K. A. and Tan, C. K., Computer Aided Tolerance Charting—A Block Box Approach, *International Journal of Production Research,* 33: 4: 1117, 1995.
9. Kusiak, A., and Feng, C. X., Deterministic Tolerance Synthesis: a Comparative Study, *Computer Aided Design,* 27: 10: 759, 1995.
10. Wei, C. and Lee, Y., Determining the Process Tolerances Based on the Manufacturing Process Capability, *International Journal of Advanced Manufacturing Technology,* 10: 6: 187, 1995.
11. Eary, D. F. and Johnson, G. E., *Process Engineering for Manufacturing,* Prentice-Hall, New Jersey, 1962.
12. Johnson, A. M. Tolerance Charts. In *Manufacturing Planning and Estimating Handbook,* McGraw-Hill, New York, 1964.
13. Wade, O. R. *Tolerance Control, Tool and Manufacturing Engineers Handbook, Vol.1, Machining,* edited by T. J. Drozda and C. Wick, Dearborn, Michigan, Society of Manufacturing Engineers, 1983.
14. Ngoi, B. K. A., Applying Linear Programming to Tolerance Chart Balancing, *International Journal of Advanced Manufacturing Technology,* 7: 187, 1992.
15. He, J. R. and Lin, G. C., Computerized Trace Method for Establishing Equations for Dimensions and Tolerances in Design and Manufacture, *International Journal of Advanced Manufacturing Technology,* 7: 210: 1992.
16. Ji, P., Determining Dimensions for Process Planning: A Backward Derivation Approach, *International Journal of Advanced Manufacturing Technology,* 11: 52,1996.
17. Ji, P., A Linear Programming Model for Tolerance Assignment in A Tolerance Chart, *International Journal of Production Research,* 31: 3: 739, 1993.
18. Lee, W. J., Woo, T. C., and Chou, S. Y., Tolerance Synthesis for Nonlinear Systems based on Nonlinear Programming, *IIE Transactions,* 25: 1: 51, 1993.
19. Lin, S. S., Zhang, C., and Wang, H. P., On Mixed-Discrete Nonlinear Optimization Problems: A Comparative Study, *Engineering Optimization,* 23: 287, 1995.
20. Irani, S. A., Mittal, R. O., and Lehtihet, E. A., Tolerance Chart Optimization, *International Journal of Production Research*, 27: 9: 1531, 1989.

21. Ngoi, B. K. A. and Kai, C. C., A Matrix Approach to Tolerance Charting, *International Journal of Advanced Manufacturing Technology,* 8: 175, 1993.
22. Yeo, S. H., Ngoi, B. K. A., and Chen, H., A Cost-Tolerance Model for Process Sequence Optimisation, *International Journal of Advanced Manufacturing Technology,* 12: 423, 1996.
23. Dong, Z. and Hu, W., Optimal Process Sequence Identification and Optimal Process Tolerance Assignment in Computer-Aided Process Planning, *Computers in Industry,* 17: 19, 1992.
24. Ostwald, P. F. and Blake, M. O., Estimating Cost Associated with Dimensional Tolerance, *Manufacturing Review,* 2: 4: 277, 1989.
25. Famer, L. E. and Gladman, C. A., Tolerancing Technology Computer-based Analysis, *Annals of CIRP,* 3: 1: 7, 1986.
26. Abdou, G. and Cheng, R., TCVAPP, Tolerance Verification in Computer-Aided Process Planning, *International Journal of Production Research*, 31: 2: 393, 1993.
27. Lin, S. S., *Mechanical Tolerancing for Integrated Design and Manufacturing,* Florida State University, 1994.
28. Wade, O. R., *Tolerance Control in Design and Manufacturing,* Industrial Press, Inc., New York, 1967.
29. Leung, H. C., Annotated Bibligraphy on Computer-Aided Process Planning, *International Journal of Advanced Manufacturing Technology,* 12: 5: 309, 1996.
30. Zhang, H. C. and Huq, M. E., Tolerancing Techniques: the-State-of-the-Art, *International Journal of Production Research,* 30: 9: 2111, 1992.
31. Zhang, C. and Wang, H. P., Integrated Tolerance Optimization with Simulated Annealing, *International Journal of Advanced Manufacturing Technology,* 8: 187, 1995.
32. Phadke, M. S., *Quality Engineering Using Robust Design,* Prentice-Hall, New Jersey, 1989.

3

Production of High Quality Parts by the Process of Grinding in Manufacturing Systems

3.1 Introduction
Control Issues Relevance
3.2 Grinding Process
History and Perspective • Grinding Wheels Materials
3.3 Grinding Process Models
Theoretical Models • Empirical Models • Force Models • Model Complexity • Grinding Process Model Development for Estimation
3.4 Grinding Model Parameter Identification
Parameter Identification via Position Control • Parameter Identification via Force Control • Multiple Input Estimation Approach • Optimizations of RLS-FF and Kalman Filters
3.5 Control
Adaptive Machine Tool Control • Grinding Control
3.6 Force Measurement
3.7 Controller Design
System Architecture: Parallel Processing and Implementation
3.8 Adaptive Grinding Process Control
Adaptive Controller Development
3.9 Summary
Nomenclature

Thomas R. Kurfess
Georgia Institute of Technology

Hodge E. Jenkins
Lucent Technologies, Inc.

Richard C. Dorf
University of California at Davis

3.1 Introduction

Grinding is an important process that creates precise parts with smooth and consistent surface finishes. It is utilized in the fabrication of many everyday products, such as bearing components, and is the final machining step in much of today's precision component manufacturing. Grinding operations are found in nearly all applications of precision materials production, for example tape heads, disk drive platens, medical implants, automobile roller and ball bearings, specialized optics, and mirrors. Since grinding has relatively low material removal rates, it can create smooth surfaces while maintaining precision tolerances. It is also a common and necessary process for machining extremely hard materials (greater than 50 Rockwell C hardness) or brittle materials (such as silicon wafers) where operations such as milling are not efficient or successful. For extremely hard materials such as ceramics or super-alloys, fixed abrasive grinding rather than milling is the preferred mode of machining, to mitigate material and tool damage (Bifano, Yi, and Kahl, 1994). Indeed, as more precise components are required in today's and tomorrow's products, grinding applications are ever increasing.

Grinding is an extensively used and important machining operation that has been stated to account for approximately 20 to 25% of all machining costs in the industrialized world (Malkin, 1989). Thus, achieving reductions in cycle time while increasing the resultant product quality are significant concerns for manufacturers as global competition increases. To achieve high quality precision surfaces in a timely manner, it is important to control the grinding process to create smooth and contiguous surfaces, obtain a desired geometry with tolerances, and limit or mitigate both surface and subsurface damage to a ground part and to the grinding wheel. To accomplish these results velocity, position, and force must be controlled during grinding. Velocity control relates to the resulting surface finish, while positional control determines the geometric accuracy. Control of the grinding force not only reduces the possibility of damage to the part, but also affects the surface finish and geometry. Variations in the normal force during grinding change the amount of deflection in the system, adversely affecting positional accuracy. Significant variations in force can also change the grinding wheel speed and the ensuing surface finish. A key component in realizing high precision and quality parts is precision control of these key parameters of the grinding process.

Unfortunately, typical industrial grinder controllers servo only position and feed velocity and do not servo grinding force levels. Existing fixed-gain, force control systems have been found to be of limited use because of process variations (Tonshoff, 1986). Indeed, a fixed-parameter model may even lead to instability or damage the part or wheel. (Additional discussion on grinding control is found later in this chapter.) It should also be noted that a grinding process model has many uses other than the development of an adaptive control system. For example, an accurate real-time model can provide an excellent means to indicate the necessity of wheel dressings for maintaining grinding efficiency, or for detecting failures in unattended grinder operations.

Control Issues Relevance

Although grinding is a precision machining operation, many problems exist that limit quality and production rates in precision grinding. Process considerations for grinding have traditionally been position and velocity accuracies, as these are critical in yielding a smooth and consistent surface finish within dimensional tolerances. In practice variations in grinding force and surface velocity can generate unwanted defects and damage to both the part and the grinding wheel (Brady, 1985; Marion, 1985). Part damage can occur in several forms: dimensional error, excess surface roughness, and material damage (e.g., surface fissures, sub-surface cracks, and thermal damage). Force variations can cause surface damage on ground parts and sub-surface damage in ductile grinding of glass and ceramic lenses, and in metals to a lesser degree. A significant portion of this sub-surface damage is not detected until the surface fails while in use. It should be noted that grinding wheel damage is also a significant cost in terms of part cost and set-up time. Super-abrasive grinding wheels are extremely expensive, having costs of more than 100 times that of conventional abrasives. Thus, to increase production rates and to improve the ground surface quality, it is desirable to control the grinding process in terms of positional accuracy, and uniformity of velocity and force level.

To limit damage to parts and grinding wheels, current industrial grinding practice is to use slower feed rates reducing productivity. Grinding is separated into two regimes, roughing and fine finishing. In roughing, relatively large feed rates (and forces) are applied to remove large amounts of material. A transition between roughing and fine finishing is performed, where the feed rate is zero, allowing a reduction of force to occur. This is called tarrying. After the time allowed for tarrying, fine finishing is commenced at low feed rates. The tarrying time is a fixed amount of time with no assurances of the desired force reduction. By controlling the force, dimensional variations and damage may be better controlled during roughing while simultaneously reducing the tarry time.

Grinding systems employing force control possess several significant advantages besides the reduction of tarry time. A related benefit is the reduction of spark-out times in a typical grinding cycle. The time

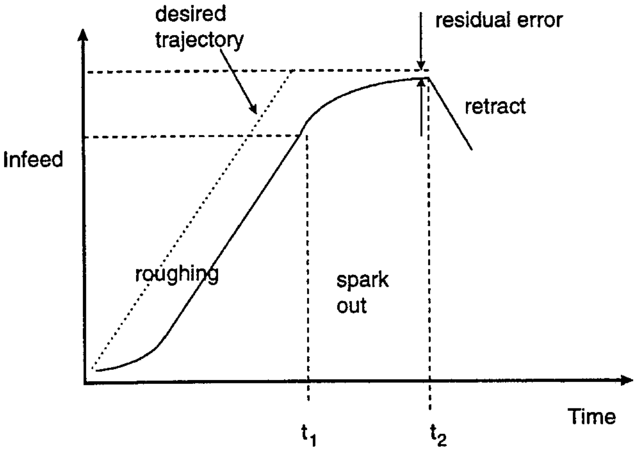


FIGURE 3.1 Typical grinding cycle time.

required to reduce the force is at the end of a grinding cycle. A typical grinding cycle is shown in Fig. 3.1. There is a trade-off between spark-out time and residual error in the ground part. Under force control a force trajectory can be followed to reduce the spark-out time, and the positional error can be reduced by providing a constant and repeatable force. The accuracy and repeatability of the imposed grinding forces translates to dimensional accuracy and repeatability from part to part.

Unfortunately, the grinding process is time-varying with changing properties and conditions in the grinding wheel, material, and existing surface profile. Therefore, a simple fixed-parameter model of the grinding process is insufficient for certain high precision applications. Thus, a controller that adapts to variations in the grinding process parameters must be developed as well as an observer (or estimator) to identify these parameters in real-time. The observer provides the controller with updated process parameters, thereby permitting the controller to adapt to process variations. This is an effective means to achieve a desired profile and surface finish with minimal damage.

3.2 Grinding Process

The tool in the grinding process is the grinding wheel, which is generally composed of two materials: grits and bonding agent. The hard abrasive grits (small hard particles) are affixed to rotating hub or axis in the general configuration of a wheel to erode material away from a workpiece as the wheel spins. Grinding wheels are formed into shape by casting and curing a slurry of bonding material with the grits. The cured wheel forms a three-dimensional matrix of grits held in place by the bonding agent. This yields a complex operation by nature, as the uneven and truly random distribution of grit surfaces of the wheel and the contact they make with the ground part are difficult at best to systematically model.

History and Perspective

Abrasive material removal (grinding) is one of the oldest machining technologies employed today, and has been utilized by people in the manufacturing of parts since the Stone Age (Malkin, 1989). A simplified grinding process can be thought of as milling using a "cutter" with a large number of teeth of irregular shape, size, and spacing (Fig. 3.2.). Each grit can be seen as a cutting tooth with varying orientation and sharpness. These grits are suspended in a bonding agent that holds the three-dimensional matrix of grits together in a form. The grinding process can vary for many reasons including: wheel sharpness, wheel microstructure, workpiece material variation, loading of workpiece material on the wheel, and other phenomena that contribute to the changing nature of the grinding process. (Loading is the phenomena of the ground material becoming attached to or embedded onto the surface of the grinding wheel.

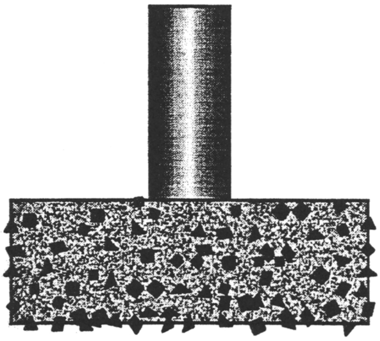


FIGURE 3.2 Grinding wheel depicting random orientation and spacing of grits.

This effect begins with the material filling in the voids around the grits, and if permitted to continue, the material can eventually cover the grits. This occurs more commonly with softer materials.)

The variable nature of the grinding process has been linked to the physical descriptions of the actions between the grits with the workpiece. Grinding is a complex operation that can be seen as three separate and concurrent process actions: (1) cutting, (2) plowing, and (3) rubbing (Hahn and Lindsay, 1971; Samuels, 1978; Salmon, 1992). The cutting action produces tangential forces that are related to material specific energy in the generation of chips. Although a small part of this energy is transferred to the chip as kinetic energy, the majority of the material specific energy is dispersed in several other ways including: friction, heat conduction, radiation, surface formation, residual stresses, etc. The second grinding process action is plowing. In plowing material is plastically deformed around the sides of dull grits, leading to ruts and uneven surfaces. No material is removed with plowing. Rubbing occurs as material loads onto the wheel or if the grits are exceptionally dull (causing attritious flat wear). Rubbing is a high friction condition with essentially no material removal and is the most detrimental of the three grinding actions. The generated frictional heat (that can yield "burning") from rubbing is dissipated into both the ground part and the wheel. If coolant is not used, thermal damage may occur. Damage created from excess forces or from plowing ruts can be hidden by the migration of material during rubbing, only to result in premature failure of the part in use. In general, all three grinding process actions occur simultaneously in varying distributions depending on the grinding wheel, material, and operating conditions. As stated previously grinding is stochastic by its nature, making detailed process analysis, modeling, and control difficult. This is also brought out by the variety and complexity of grinding models presented later.

Grinding Wheels Materials

Natural corundums (aluminum and iron oxides) were used as one of the first grinding materials by the early Roman and Greek civilizations. The use of corundum continued throughout the industrial revolution and into the nineteenth century. It was not until the inexpensive production of aluminum in the late nineteenth century that man-made, more pure aluminum oxide was available for use. By the beginning of the twentieth century grinding wheels were being constructed of synthetic aluminum oxide and silicon carbide. Common bonding agents were vitrified resin, rubber, and shellac. The selection of the bond depended on the hardness of the material to be ground and the amount of self-dressing desired. These synthetic aluminum oxide and silicon carbide wheels are found and used in production today.

In addition to aluminum oxide and silicon carbide, super abrasive materials are employed in today's grinding wheels, such as CBN (cubic boron nitride) and diamond. The use of diamonds is mainly limited to the grinding of ceramics and glasses. CBN is the newest of the widely-used fixed abrasive materials (aluminum oxide, silicon carbide, and CBN), and has a large number of advantages over conventional corundum including greater hardness, better wear resistance, and lower potential for thermal damage risk. In fact, the hardness of CBN is surpassed only by that of diamond. Despite these benefits, much of today's grinding is accomplished with corundum, as corundum (aluminum oxide) is relatively inexpensive compared to CBN and diamond wheels. The current cost of CBN is several orders of magnitude higher than aluminum oxide. Its G-ratio (ratio of material removal to wheel wear) is also about the same order of magnitude greater, offsetting its cost by its longer life. However, the use of CBN wheels have other complications such as longer "break-in" periods after dressing due to its greater hardness and wear resistance. During the break-in period the surface roughness generated can be larger than desired. Sharp grits located on the wheel's high spots can create ruts. CBN requires longer time for wearing away of the initial sharp high spots on newly dressed wheels, that can result in relatively lower quality of parts after dressing.

Although the grinding wheel materials are generally the hardest materials known to man (2100 to 7000 on the Knoop scale, where 60 $R_c = 740$ Knoop), during grinding there is wear of the wheel. The wearing of flats on grains with corundum is the predominant form of wheel wear and leads to varying process performance. In many cases wheel wear is intentional; as grits dull they fall out and expose new underlying sharp grits, and the wheel is continuously dressed. This process is known as self-dressing. All wheels require truing and dressing. Truing places the desired profile on the wheel, while dressing exposes grits resulting in the desired wheel sharpness. Wheels are trued and redressed as needed, based mainly on the characteristic of the resulting part, and not a measure of the wheel itself. (Although the grinding process model may provide a means to indicate a need for wheel dressing.)

3.3 Grinding Process Models

In order to better understand, predict, and control the grinding process, a variety of process models have been developed. These models encompass many aspects of grinding. Most models address a particular process attribute or variable, such as force, energy, temperature, surface finish, or topography. Some models combine several process effects. In all cases the purpose is to relate causes and effects, and to provide a better understanding for using grinding more productively and more accurately. A summary of such grinding models is presented by Tonshoff (1992).

Modeling of the grinding process (as well as other processes) can generally be divided between empirical and theoretical models. Both types of models represent the physical process and have a basis in physical laws and scientific knowledge. Modeling generally involves a trade-off between the accuracy of the model and the difficulty (or physical possibility) of obtaining the necessary information or parameters. In general, theoretical grinding models are based on idealizations of the grinding process. The idealizations include viewing grits as sharp, v-shaped single point grains with the average (estimated) number of active grains per unit area. Models have been developed using analogies to other material removal process. Indeed, many grinding models determine the cutting forces based on "chip" width, depth of cut, and the material specific energy similar to milling. It is important to note that many of the chip model theories used in other machining operations such as milling, turning, and drilling are not entirely relevant to grinding, as discussed by Malkin (1989).

Theoretical Models

Theoretical models are purely derived from the physical laws, process attributes, process variables, and physical assumptions. Of course, the theoretical models must be verified by experiments. These theoretical models are often preferred, as generalization of the process is contained directly in the formulation. Thus, by replacing the constants and boundary conditions, the theory can be transported to different situations. However, in complex multi-contact point processes, such as grinding, the inability to measure and account for all the necessary conditions at a sufficient detail generally precludes the use of a purely deterministic model. So averaged or statistical values of process parameters such as the number of cutting points in contact with the workpiece are used in theoretical grinding models, as opposed to the deterministic parameters used to develop chip models in milling, that address all cutting points.

Grinding wheel grit-related properties and the effective contact length are important parameters in theoretical models. Malkin (1989) and Verkerk (1975), among others, have conducted extensive work in determining the effective contact length (or contact areas) for various grinding operations. As expected

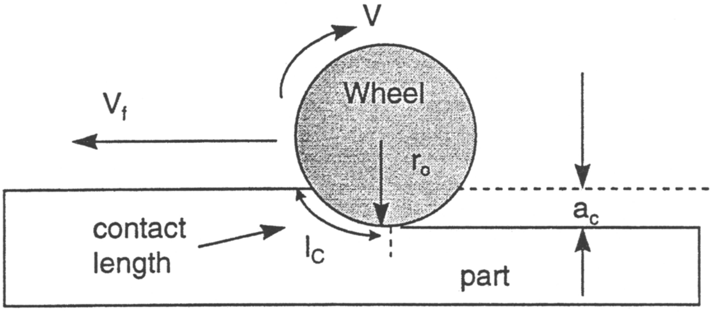


FIGURE 3.3 Straight surface grinding.

the contact length depends on the type of grinding being performed, the geometry of the wheel, the workpiece, the applied force, and the depth of cut. For straight surface grinding the static contact length, l_c, is depicted in Fig. 3.3, where V is the surface speed of the wheel, V_f is the feedrate, and a_c is the engagement depth.

The geometric relationship for the contact length in straight surface milling for an undeformed wheel and workpiece is given in Eq. (3.1).

$$l_c = (2a_c r_o)^{1/2} \tag{3.1}$$

It should be noted that the real contact length is larger than this geometric relationship, owing accountability for loading and wheel speed (Verkerk, 1975).

Research into the grinding contact lengths and contact areas parallels that of Kalker (1966, 1968) and Vermeulen and Johnson (1964) for non-spherical contact of two bodies. These researchers examined the two-body roll-slip phenomena, generally applied to wheeled-rail vehicle dynamics. The slip motion is referred to as longitudinal creep in this context. It is this relative slipping kinematics that is related to surface grinding and cylindrical grinding. Extension of these rail-wheel slip/roll dynamics is not necessary in a face grinding operation. However, it has application in two-dimensional surface grinding, and three-dimensional grinding, where geometry can be more complex. In these cases Hertzian theory is used to determine the contact area as part of the mathematical solution to evaluate relative motion and tangential forces.

There are limitations in using Hertzian contact theory in grinding; these are detailed by Rowe and Morgan (1993). A contact model based on Hertz's contact theory generally requires that: (1) the contact surfaces are smooth, (2) the deformations are elastic, and (3) material removal is not addressed. Rowe points out that the Hertzian approach is valid at high loads, where the grinding pressure is distributed more evenly. However, at lower pressures the distribution varies significantly and the Hertzian model is not appropriate.

Relations for the contact lengths and areas have been developed for all common grinding operations including cylindrical grinding, centerless grinding and face grinding. For face grinding the calculation of contact area is direct, and not dependent on the depth of cut, as illustrated in Fig. 3.4.

Theoretical grinding models for calculating material removal are based on the specific energy of the material and energy distribution of the applied power. The fundamental relationship derived in machining is the specific energy (material removal rate/applied power) as described by Mechant (1945). Any relationship for material removal must account for this energy relationship.

In general the applied power can be calculated as product of the tangential force, F_T, and the relative surface speed of the wheel, V.

$$P = F_T V \tag{3.2}$$

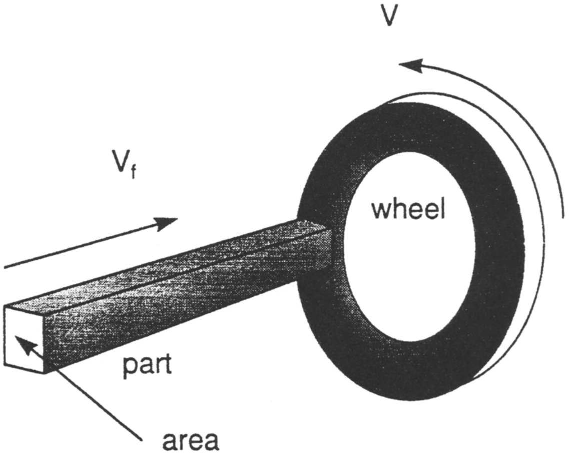


FIGURE 3.4 Face grinding.

The material specific grinding energy, u, is defined as the applied power divided by the material removal rate, Q.

$$u = \frac{P}{Q} \quad \text{with} \qquad Q = V_f \times \text{area of cut}. \tag{3.3}$$

The area of cut is typically the product of the working engagement and the grinding area. Of course all of the expended energy in grinding is not used in cutting material. As mentioned previously, three separate actions are occurring (cutting, plowing, and sliding/rubbing), and the applied energy is distributed among these actions (Eq. (3.4)). Also, the momentum contribution is generally small in comparison with the other terms (typically less than 1%) and is usually neglected.

$$u = u_{\text{cutting}} + u_{\text{rubbing}} + u_{\text{plowing}} + u_{\text{momentum}}. \tag{3.4}$$

Likewise, the tangential force is composed of several distinct contributions, the tangential cutting force, $F_{T,C}$, the frictional or rubbing force, $F_{T,R}$, and the tangential force attributed to the plastic deformation in plowing, $F_{T,P}$ (momentum is neglected).

$$F_T = F_{T,C} + F_{T,R} + F_{T,P}. \tag{3.5}$$

The cutting force is related to the working engagement, the material specific cutting energy, and chip formation. The rubbing force is proportional to the amount of wear and dulling of the grits on the grinding wheel. Similarly, the normal force is the sum of the normal cutting, rubbing, and plowing force components.

$$F_N = F_{N,C} + F_{N,R} + F_{N,P}. \tag{3.6}$$

Material removal (cutting) in grinding occurs via a shearing action generated in chip formation. This is common to other machining processes. Applying Eqs. (3.3), (3.4), and (3.5) the cutting component of the specific energy can be defined as

$$u_{\text{cutting}} = \frac{F_{T,C} v_S}{Q}. \tag{3.7}$$

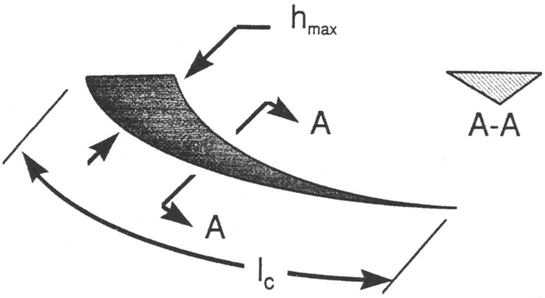


FIGURE 3.5 Chip thickness model for straight surface grinding.

The other specific energies attributable to rubbing and plowing are defined similarly. The rubbing energy is attributable to the wear flats on the grits, and is dependent on the frictional coefficient and the wear flat area. Plowing occurs in the deformation of the material without any material removal. Some plowing action is theorized to occur in the initial entry of a grit into the material before the actual chip formation occurs. This contributes to the chip-size effect on the required specific energies.

Although several chip thickness-based models have been developed, it is difficult to estimate the contribution of each of the three grinding actions to the overall process. Likewise it is not realistic to separate the action-specific forces and determine the components of the total specific energy (Malkin, 1989). This is evident by the fact the specific material energies have been found to be higher for grinding than for other machining processes, although the specific energy is thought of as a material property. Several hypotheses have been proposed based on observations that there is a correlation between the chip size and the needed cutting force. Indeed, there has been found to be a considerable size effect in grinding (Marshall, and Shaw, 1952). The first hypothesis for this occurrence is that smaller chip sizes promote increase stress flow, reducing the stress concentration of larger chip sizes alone (Backer, Marshall, and Shaw, 1952). Secondly, the grinding force is distributed (See Eq. (3.5)) among the three basic actions, cutting, plowing, and rubbing, thereby requiring higher applied forces than would otherwise be estimated by the cutting action alone. Again the determination of the allocation of the forces due to cutting, plowing, and rubbing is difficult, so both hypotheses are reasonable but not completely proven.

Most theoretical grinding models use undeformed chip thickness theory and contact lengths to estimate grinding forces, material removal, and topology models. The undeformed chip thickness model has been held as the primary metric of grinding conditions (Kumar and Shaw, 1981). Since grinding is a multi-point contact operation with intermittent cutting and other process specific complications, average values of grain dimensions and contact lengths are used in the chip thickness theory, drawing upon a milling analogy for the shape of the chip. For straight surface grinding the chip formation and thickness are similar to milling, as illustrated in Fig. 3.5. Although the cross-section depicts a triangular shaped grit, grit cutting profiles are varied and change with attritious wear. Note in the figure that $h_{\max}$ refers to the maximum undeformed chip thickness while l_c refers to the contact or engagement length.

Undeformed thickness models are provided by several sources including the extensive work and summary by Malkin (1981). A chip thickness model is presented in Eq. (3.8) for straight surface grinding.

$$h_{\text{MAX}} = \left[\frac{4}{Cr}\frac{V_W}{V}\frac{a_c}{\text{whell diameter}}\right]^{1/2}. \tag{3.8}$$

In this equation, C is the number of active cutting points per unit area, r is the ratio of the theoretical chip width to height, a_c is the engagement, V_W is the wheel speed, and V is the surface speed.

Other chip thickness models in grinding are based on relating the volumetric material removal rate to the average material removal rate per average active grit. Using such a basis, the theoretical undeformed chip thickness in face grinding for a constant triangular cross-section is

$$h_{\mathrm{MAX}} = \left[\frac{\sqrt{3}}{Cr}\frac{V_W}{V}\right]^{1/2}. \tag{3.9}$$

A typical relationship for estimating material removal based on the average or maximum chip thickness and contact length is given in (10) (Liang, 1994).

$$Q = Va_pCr_gh_{\mathrm{MAX}}^2\frac{l_C}{4} \tag{3.10}$$

where r_g is the average grit aspect ratio, a_p is the grinding contact area, and the other variables are previously defined.

Despite the advances in theoretical chip models, process variation (caused by wheel and part material variations, wear, and loading) the main thrust of grinding process modeling has been directed towards empirical models, especially for real-time applications. An interesting exception to this is precision engineering work where a single-point diamond tip is used to cut surfaces at a nanometric level (Dresher and Dow, 1990; Fawcett and Dow, 1991). However, most of this nanometric work is limited to research focusing on precision ductile regime turning and grinding of monocrystalline or amorphous glass optics.

Empirical Models

Although theoretical models have generalizable equations, potentially applicable to a multitude of situations, their many unknown variables limit their accuracy and appropriateness for control applications. Therefore, empirical models dominate grinding process modeling, especially in the calculation of forces and material removal. There are several reasons for this, including the fact that empirical models can be developed with relative ease. Experimental results can be used to determine process models, parameters can be based solely on the data correlations. However, the quality of such empirical models can vary and can be limited, depending on the specific process conditions. Also it is important to establish cause and effect, rather than correlating effects. Thus, the most successful models, whether theoretical or empirical, have a relation based on an underlying physical principals as well as empirical correlations.

As the grinding process is time-varying, time-invariant empirical models have limitations to their applicability. Generally, such models are specific to the type of grinding, material, wheel type and wheel conditions. With continued grinding, grits dull or drop out, the location and orientation of newly exposed grits varies, and the wheel can load. With all these potentials for change, the conditions and time duration of empirical model validity is a significant concern.

Empirical models for grinding and abrasive machining have been dated as early as 1927 (Brown, 1990). Many empirical models have been used to describe various concerns of the grinding process. Grinding models, focusing on different process aspects, can be separated into several categories: (1) force and energy, (2) temperature, (3) roughness, (4) topography, (5) vibration and chatter, and (6) wear.

Force Models

The link between the normal contact force and the material removal rate has been established in empirical models by several research groups (Salje, 1953; Shaw, 1956; Salmon, 1992; Brown, 1990; Hahn and Lindsay, 1971). The selection of an appropriate mathematical model of the physical grinding process for real-time control is the objective of this chapter; therefore, further discussion of empirical models will be limited to force models. Hahn and Lindsay (1971) are generally regarded as publishing one of the first modern models relating MRR to normal forces in grinding,

$$Q = \Lambda_W(F_N - F_{TH}) \tag{3.11}$$

where the variable (Q) represents the material removal rate, and is expressed in volume per unit time, and F_N and F_{TH} are the normal force and threshold normal force, respectively. The threshold normal force, F_{TH}, is the lower limit of the normal force, below which no significant amount of material removal takes place. The material removal coefficient (Λ_W) is a function of many parameters including friction coefficient, material specific energy and wheel velocity. As several of the measures required to determine Λ_W are not easily obtained, the coefficient is usually determined directly from experimental results.

Other early grinding research was conducted in the 1950s and 1960s by Shaw and others, as well as Preston in 1927. Preston observed the relation between material removal and the abrasive power (the product of the normal force and velocity) in loose abrasive polishing of optics. The Preston equation (3.12) has been previously used to define the relationship between the normal grinding force, wheel contact speed, and the MRR without directly addressing friction and specific energy (Brown, 1990). The empirically-determined coefficient for the grinding model based on the Preston equation can be expeditiously found. (Note the form is quite similar to that of Eq. (3.11).)

$$Q = K_P F_N V \tag{3.12}$$

The coefficient K_P correlates the normal grinding force to expected removal rates, given the relative surface velocity, for specific materials and conditions of the workpiece and wheel. F_N is the normal force between the part and the grinding wheel, while V is the relative velocity between the surfaces of the grinding wheel and the part.

Model Complexity

The model developed in Eq. (3.11) has been shown to be adequate for real-time estimation and adaptive control of the grinding process. However, other more complex formulations have been developed for grinding force models (Salje, 1953; Shaw, 1956; Konig, 1973). All these grinding force models have a basic formulation (Konig 1979, 1989) where the normal force (or tangential force) is related to various combinations of the working engagement, a_e, cutting speed, V_c, feedrate, V_f, wheel diameter, d_e, and specific energy, u. The general form of the relationships is given in Eq. (3.13). In this equation the exponents, e_i ($i = 1, 2, 3, 4$), vary depending on the author. In some models other terms are also present in the basic equation. For example, Salje uses the shear strength is a parameter, while a model by Ono makes use of the average grain distance. The coefficient, K_o, and the exponents are determined experimentally for particular situations.

$$F_N \quad \text{or} \quad F_T = K_o \left(\frac{V_{\text{feed}}}{V_C} \right)^{\text{el}} a_e^{e2} d_e^{e3} u^{e4} \tag{3.13}$$

As previously mentioned, these models are based on the chip thickness theory and the grinding contact length, as well as a power balance. However, the exponents and coefficients of Eq. (3.13) are purely empirical and depend on the specific grinding conditions and materials.

More recent grinding force models also present a similar formulation to that of Eqs. (3.11) and (3.12), as given by Malkin (1989) in Eq. (3.14)

$$Q = \frac{F_{N,C} V A}{k_l u_C}. \tag{3.14}$$

From this it can be seen that K_P of Eq. (3.12) is identical to $(k_1 u_c)^{-1}$ term in Eq. (3.14). This is also close to the form of the model proposed by Coes (1972) in Eq. (3.15) accounting for wheel wear, q_w, and dulling by attrition, q_a.

$$Q = \left(\frac{q_w}{q_a + q_w} \right) K F_N V \tag{3.15}$$

Empirical force models must account for different materials and grinding wheels. Thus, the use of many of these empirical models requires significant effort in the determination of the exponents, as well as coefficients when new conditions arise. Much work has been conducted in this area for such models (Hahn and Lindsay, 1971; Konig, 1974)

Another recent, yet more complex, attempt to improve grinding models previously presented has been met with limited success. Younis (1987) provided additional terms in his model in an attempt to account for the frictional coefficient at the grit tip, the stress concentration from ploughing, and wheel loading. This was performed in an effort address the separation of energy and forces related to rubbing, cutting, and ploughing. The predictive results of the model still show significant (20% to 50% error) discrepancies with experiments, limiting the value of these predictive models. Thus, this discussion concentrates on improving the real-time parameter estimation of simple models with frequent update from sensors.

Grinding Process Model Development for Estimation

A recent grinding process model developed and used to relate material removal rates to the measured normal force and surface speed is presented and validated by Ludwick et al. (1996). The grinding process model is selected based on a combination of the grinding model developed by Hahn and Lindsay as well as the Preston model given in Eqs. (3.11) and (3.12), respectively. The resulting, two-parameter model is given below in Eq (3.16).

$$Q = K_P(F_N - F_{TH})V; \qquad F_N \geq F_{TH} \tag{3.16}$$

The variables in Eq. (3.16) represent the same parameters described for Eqs. (3.11) and (3.12).

The appeal of this model form for determining material removal is its simplicity. This is especially desirable in applications of real-time parameter estimation. The material removal rate is linearly related to the product of the normal force and surface velocity, in a traditional power relationship. The coefficient, K_P, and the threshold force, F_{TH}, in this model are determined experimentally. These coefficients represent the current condition of the specific grinding wheel and workpiece. Thus, estimations or calculations to separate rubbing, plowing, and cutting effects of the process are not needed. The variables described in this model are depicted in terms of the actual physical system

It should be noted that although Preston's original work was related to abrasive machining and polishing of optics, the model used in this work is similar to the model formulation of Hahn and Lindsay, Malkin, Coes, and others. All the presented empirical models relate grinding forces (either normal or tangential) to a product of various material removal rate parameters such as wheel speed, engagement, equivalent diameter, and peripheral speed. The form of these force models is similar to that of Eq. (3.16) selected for this research, with the main variations being the number of parameters and their exponents. Again, all of these models are founded in a basic power relationship (chip formation) for determining material removal.

Equation (3.16) can be written using the wheel feed velocity and the contact area as:

$$\dot{x}_f = \frac{(F_N - F_{TH})K_P V}{A}. \tag{3.17}$$

Equation (3.17) is rewritten to relate the normal force to the grinding wheel feed velocity

$$F_N - F_{TH} = \dot{x}_f \frac{A}{K_P V}. \tag{3.18}$$

Upon examining Eq. (3.18), it should be apparent that the grinding process dynamics may be thought of as being equivalent to a viscous damper as the feed rate (velocity), $\dot{x}_f$, of the grinding wheel through the part is linearly related to the normal force, F_N (minus the threshold force). This assumes that part and grinding wheel remain in contact and is compatible with the physical observations of the process.

A schematic diagram of a typical grinding process for position controlled grinding is depicted in Fig. 3.6. A block diagram of the grinding process based on Eq. (3.16) is seen in the exploded view of the process in

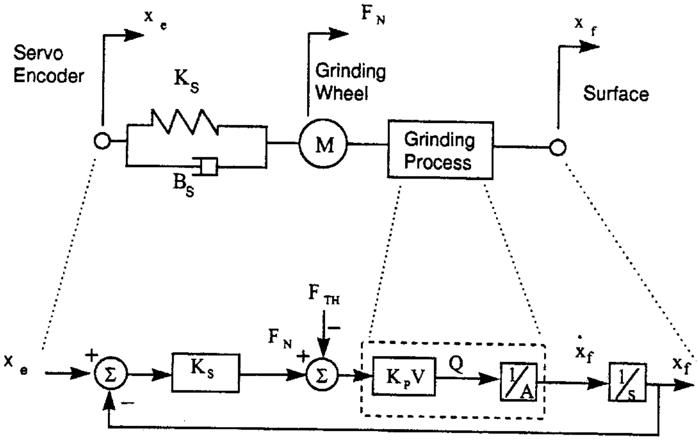


FIGURE 3.6 Block diagram of model of the system with the grinding process.

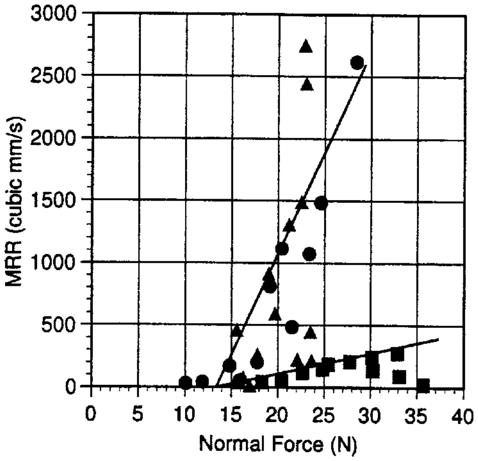


FIGURE 3.7 MRR vs. Normal grinding force (3 Set-ups for 1020 steel).

the lower section of the figure. Here the relationship between the measured output of the position loop, (x_e) and the actual ground surface position, x_f, for a plunge grinding is seen. Note the interaction between the servo controlled position measurement, x_e, and the workpiece may be thought of as a spring, K_S, and damper, B_S. Higher order system effects of the grinding wheel are generally neglected in this approach. It can be readily seen from examining Fig. 3.6, that the grinding process affects the overall system response and any attempt to control the grinding normal force.

The process model of the complete grinding system is given in the box with the dotted line in Fig. 3.6. It is apparent that the part surface in contact with the grinding wheel is continuously being ground away, thus its position is always changing. Therefore, the position of the part surface (x_f) is subtracted from the controlled position (x_e) in order to determine the relative compression of the system. When this relative compression is multiplied by the effective impedance of the servo stage/force sensor/grinder ($K_s + s\ B_s$), the normal force (F_N) between the part and the wheel is found.

The normal force, less the threshold normal force, is entered into grinding force model equation (multiplied by $K_P V$) to determine the expected material removal rate (Q). This material removal rate is assumed to be uniform over the contact area, and so dividing the material removal rate by the contact area yields the velocity at which the ground surface is advancing ($\dot{x}_f$). Integrating this value produces the ground surface position, x_f. Note that the compliance of strain gage force sensors has been shown to be a significant part of the contact stiffness in applications where they are used (Book and Kwon, 1992).

Using Black's Law, the process dynamics in Fig. 3.6 may be reduced to an algebraic equation. This equation is simplified by assuming the effective damping of the servo stage and the structural damping are negligible (*e.g.*, $B_s \to 0$). This assumption is made possible by the low speeds at which the grinding wheel plunges into the part. Neglecting the effect of damping yields the following first order equation

$$\frac{x_f(s)}{x_e(s)} = \frac{K_P K_S V/A}{s + K_P K_S V/A}. \tag{3.19}$$

3.4 Grinding Model Parameter Identification

Two types of experiments can be conducted to estimate the model parameters, F_{TH} and K_P, of Eq. (3.16). These parameters are identified by conducting position controlled grinding experiments and force controlled grinding experiments. The mathematical basis for identification in both types of experiments is outlined in the following paragraphs. The purpose of these experiments is to validate the appropriateness of the selected grinding model.

Parameter Identification via Position Control

The first type of experiment uses position control of the grinding system to provide position step inputs. The response of the grinding wheel, $x_f(t)$, to the position step input of magnitude (X_e), is given in Eq. (3.20).

$$x_f(t) = \mathcal{L}^{-1}\left[\frac{K_P K_S V/A}{s + K_P K_S V/A}\left(\frac{X_e}{s}\right)\right] \tag{3.20}$$

Taking the inverse Laplace transform yields an exponential relationship

$$x_f(t) = X_e(1 - e^{-(K_P K_S V/A)t}). \tag{3.21}$$

Similarly the transfer function from the controlled encoder position, x_e, to the predicted grinding normal force, F_N, (between the part and the grinding wheel) can be formulated

$$\frac{F_N(s)}{X_e(s)} = K_S\left(\frac{s}{s + K_P K_S V/A}\right). \tag{3.22}$$

Again, the normal force response to a step position input of magnitude (x_e) is exponential in nature, and is given by Eq. (3.23).

$$F_N(t) = K_S(e^{-(K_P K_S V/A)t})X_e \tag{3.23}$$

Based on Eqs. (3.22) and (3.23) for an ideal step function, the position response of the grinder (and ground surface) is a first order system with an exponential rise from zero to a maximum value. The imposed normal force will start at the maximum value and decay exponentially until it reaches the threshold force, F_{TH}. The exponent, ($K_P K_s V/A$), of Eq. (3.21) is estimated in the position control experiments off-line by the use of an auto-regressive, least squares parameter estimation technique (ARX, AutoRegressive eXternal input, Ljung, 1987).

After identifying the exponent, ($K_P\, K_s V/A$), of Eqs. (3.21) and (3.23), K_P is determined via substitution of the known or calculated values of the constants. The residual value of normal grinding force when there is no appreciable grinding is the estimate of the threshold force, F_{TH}.

The determined grinding model appears to be satisfactory over small time spans. However, the use of a constant coefficient, K_P, may to yield inaccurate model results with increasing time. This is expected due to the known variable nature of the grinding process. If the model grinding parameters can be frequently updated, the long-term accuracy of the model should be improved.

It should be noted that the threshold force is found to be fairly constant for a given workpiece material and grinding wheel. Grinding experiments have also indicated that the coefficient, K_P, of the selected grinding model cannot be assumed to be constant for accurate modeling of the grinding process. Thus, real-time estimation of K_P is applied. The first attempt at this is to calculate the coefficient in real-time and use low-pass filtering to remove the noise due to numerical differentiation. The effectiveness of this direct approach is limited, depending on the process variation.

Parameter Identification via Force Control

In a second experimental method using the same model, the normal force, F_N, is maintained at constant levels to determine K_P and F_{TH}. From Eq. (3.24) it can be seen that maintaining a constant normal grinding force (along with constant surface speed and contact area) results in a constant MRR. Thus, an average value of the coefficient, K_P, may be experimentally found by achieving a constant normal grinding force and rewriting Eq. (3.18) to the following form:

$$K_P = \frac{\dot{x}_f A}{(F_N - F_{TH})V} \cong \text{constant.} \tag{3.24}$$

A series of experiments using several different force levels can be performed to determine average values for both F_{TH} and K_P.

Multiple Input Estimation Approach

Clearly, correlations exist for sensory data other than normal force, such as wheel speed or input power. Thus, it is proposed that sensor fusion with multi-input estimation may be employed to extract and utilize potential additional information from each sensor. Therefore, sensor fusion and multi-variable estimation using recursive techniques such as Kalman filtering and other recursive methods should provide an approach for data integration. These techniques should be able to extract this type of embedded information and improve the estimate of K_P.

In noisy processes (such as grinding) where modeling accuracy has limitations, it is often necessary to use additional sensors to improve the estimate of a model parameter. Indeed, this approach has been successfully applied in research areas where modeling and direct sensing are difficult or impossible (Jenkins, 1996). In this research several indirect measures of wear have been more successful at grinding model parameter estimation than any single sensor alone. This indicates that multiple sensor methods can be more reliable than any single sensor method. (The additional sensors also provide an assurance of estimation accuracy in the presence of a sensor fault.) Multi-sensor approaches (such as Kalman filtering, recursive least squares methods, neural networks, basis functions, multi-variable regression, etc.) have yielded encouraging results toward improving the on-line estimation of wear. (It should be noted that not all of these methods may be appropriate for the grinding model parameter estimation problem.) Applied to grinding, these multi-sensor techniques can integrate normal force, tangential force, velocity, and power data as input for grinding process model parameter estimation. For example, in the grinding model presented by Hahn and Lindsay in Eq. (3.11), the unknown parameters, Λ_w and F_{TH}, are process model states that are to be estimated. In grinding many correlations exist with sensory data such as wheel speed, tangential force, or power. Jenkins and Kurfess have shown that additional sensor fusion with multi-input estimation may be employed to extract and utilize any additional information from each sensor to provide a more accurate model and robust controller.

To remove noise and random machine vibrations from the model parameter estimation, traditional low-pass filtering of the sensor data is applied. This can provide adequate filtering for parameter estimation results in many applications. However, in real-time applications of control and prediction, recursive methods are generally needed. Many recursive identification methods are limited to single input-single-output systems. For multiple sensor input data, recursive methods are limited to a general recursive least squares (RLS) form. Within this general form are several applicable adaptations, including RLS with a forgetting factor (RLS-FF), gradient approaches, and the Kalman filter. (These adaptation methods may even be combined as shown by Pappas et al., 1988.)

The Kalman filter is the most general of the multiple input RLS methods used to estimate time varying parameters (Ljung, 1987). Much of the current estimation research has focused on using Kalman filtering for state (parameter) estimation. It is an optimal estimator for time varying parameters, minimizing the state error variance at each time. The Kalman filter is optimal in the sense of estimating a pure signal from one contaminated with white or Gaussian noise. Similarly, the Weiner filter minimizes the sum of the state error variance over time, and thus, it is mainly suited for stationary parameter estimation. If appropriately applied, the Kalman filter has been demonstrated to yield better results than the many of the other RLS based methods (Danai and Ulsoy, 1987).

When all the parameters of the pure signal and noise are known, the (optimal) Kalman filter can be easily obtained. It has been shown that the exponential convergence of the Kalman Filter makes it robust to modeling errors and faulty covariance statistics (Ydstie, 1985). However, the Kalman filter has been noted to be difficult to apply because of the choosing of the appropriate noise intensities, when they are functions of time. Poor choices of noise covariances often cause filter divergence to occur. Adaptive noise estimation schemes can eliminate the necessity of choosing *a priori* noise covariances. However, these techniques have also been found to be difficult to implement, and when the noise is non-Gaussian the schemes may fail (Ydstie, 1985; Ulsoy, 1989). Previous solutions addressing this difficulty have included the use of parallel Kalman filters with variable memory length using variable forgetting factors (Wojcik, 1987).

It is also important to note that in real-time applications the matrix inversion requirement of the multiple input Kalman Filter is computationally intensive. This has lead many authors to avoid the Kalman filter for multiple input/multiple output (MIMO) systems. An improved Kalman filter approach for multiple inputs is that taken by Maybeck and Pogoda (1989) who used a weighted average of several single input filters to determine a parameter.

The most widely used method for estimation of a model is a least squares fit. Here data are fit to a model form to have the minimal sum of the squares of the difference between the actual data and the predicted model. This technique is readily modified for real-time, recursive calculations and multiple sources of input.

Ulsoy (1989) has identified two such special cases of RLS that attain fairly good results for time-variable parameter estimation. Many RLS approaches rely on windowing techniques (or forgetting factors) to improve estimation of transient behavior. In some cases, this may cause the asymptotic parameter error variation to be large (Wiberg, 1987), but this is generally not the case. In general RLS methods can be obtained as a special case of the Kalman filter. Therefore, RLS methods including optimal estimators may be best for the task of estimating transient responses with multiple input sources. More details on the RLS-FF and Kalman filter approaches are provided in texts on digital control and digital signal processing.

Other methods for multi-variable input for estimation include basis functions and neural networks. The basis function (BF) method uses a series expansion in terms of a limited number of known functions that form a basis (such as a discrete Fourier Transform). The coefficients of the functions can be found by applying RLS methods (Ljung and Soderstrom, 1983). BF methods work best when the number of functions is limited and when a parameter variation can be expressed in few terms. It has also been demonstrated that basis functions are not always robust, and require some prior knowledge of the parameter variation to satisfactorily represent the data set. Since the time varying response of the grinding model coefficient, K_p, is not known *a priori* it is difficult to limit the type and number of functions needed to span a basis. Therefore, BF methods are not employed in this research effort.

Attempts at relating measured variables to models have included the use of pattern matching techniques embodied as neural networks (Rangwala and Dornfield, 1990; Purushothaman and Srinivasa,1994). Neural network approaches require no pre-determined model structure, as knowledge and relational patterns are trained into the network. A multi-level neural network structure with non-linear activations can store knowledge and attain a higher level feature abstraction. Back propagation of errors during training using a learning rule is used to create relations in the network. As this knowledge is stored in the connections, it can be utilized readily in on-line classification. More detail on neural networks can be found in Rummelhart and McClelland (1986).

Despite all the advantages of neural networks, this approach has severe restrictions and limitations for use in real-time estimation. The number of training cases needed to bound the set of possible solutions for the grinding process parameter may not be practical (or even known) with grinding. Training data are dependent on the particular setup in grinding, and are not usually known *a priori*. Also training of a back-propagation network can take thousands of repetitions to generalize and complete learning to an acceptable error level. Thus, the training set size and learning cycles may be too large to be practical. Linear networks may be trained adaptively but cannot produce results significantly different from linear RLS methods. Therefore, neural network methods are not examined for real-time, multi-sensor estimation in grinding.

Based on these attributes the approach taken for the development of real-time multi-sensor techniques to improve the estimate of the grinding model parameter is generally limited to the recursive techniques including a windowed Kalman filter and recursive least squares with a forgetting factor (RLS-FF).

Optimizations of RLS-FF and Kalman Filters

Window size variation is critical for the estimation techniques used in adaptive control of precision grinding systems, and a key metric to determining the appropriate window size is the displacement error covariance. The window size represents a trade-off between accuracy and timeliness of the data. If the window is too large, the data can be old and less useful. If the window is too small, the statistical variances due to noise are too large. Depending on the sensor sampling rate and the process variation spectrum, there should be a preferred sampling window balancing the timeliness of sensor data and a meaningful statistical representation of process. Thus, if the frequency content of the input data increases, it is speculated that relatively smaller windows are required to have a minimal estimation error.

Similar effects are expected for the both the Kalman filter and RLS-FF technique. In the RLS-FF estimation scheme smaller forgetting factors (larger discounting with time) are needed where higher frequency content is important in the parameter estimation, much like the shorter windows for the windowed Kalman filter.

For multi-dimensional systems, there are several inputs and outputs, one for each axis. Cross-coupling of the inputs and outputs can occur. For example, if the desired outputs of a two-dimensional system are force and velocity, an increase in force on one axis may slow the velocity of the other axis. Treatment of two-axis systems with cross coupling can be found in Jenkins, Kurfess, and Dorf (1996). The system described in that research utilized one axis to control velocity while the other axis provided force regulation. This approach is applicable to traverse grinding. For plunge grinding using only a single axis, multiple axis coupling is not an issue.

3.5 Control

To control the grinding process means to control the position, feed velocity, and normal force, as these process measures relate to the quality of the final ground part in terms of dimensional accuracy, surface finish, and mitigation of damage, respectively. Currently, industrial grinding systems only control the grinding process in terms of position and velocity. Limitations in modeling, control, and computer hardware have impeded the progress of force control in grinding.

Generally fixed-gain controllers (such as PID-type controllers) are preferred for applications because of their simplicity in implementation. These controllers are easily tuned using simple approaches. Many commercial motion control applications for position and velocity loops have self-tuning programs using Ziegler-Nichols tuning or other algorithms. However, many processes or systems vary sufficiently to limit the usefulness or stability of fixed gain controllers. In these cases adaptive control may be applied to maintain stability and improve performance.

Adaptive Machine Tool Control

Traditionally the control of a machine tool, such as a grinder, consists of servoing the tool position and feed velocity. However, there are many important grinding process variables and attributes that affect part quality. The process forces, applied power, and the material removal rate during grinding must also be considered to effectively produce parts without damage (as discussed earlier). Another consideration is that grinding is a noisy and time-varying process that is affected by changing properties and conditions in the grinding wheel, material, and existing surface profile.

A real-time grinding process model and adaptive controller are needed to provide a robust force controller design suitable for creating precision surfaces in a timely manner. Control of the normal grinding force will allow manufacturers to attain higher production rates on precision parts without damage, and give the ability to grind for finishing with reduced part deviations from deflections induced by force variations.

Recursive parameter estimation techniques employing forgetting and windowing may be used to integrate multiple sensor inputs and optimize the parameter estimation. The resulting parametric model is used with a suitable adaptive control scheme to reduce the effects of the identified process variation.

As previously discussed, the basis for modeling a machining operation is a power relationship between forces, speed, process variables, and the material removal rate (MRR). Most machining processes (milling, turning, drilling, grinding) have this type of relationship or model for estimating the MRR from other process variables. This is a necessary step in the design of a successful adaptive process controller.

Adaptive controllers generally follow a similar approach to control a system. Estimation of the current or future states is used to determine updated controller gains based on either a design calculation or in reference to a model. Thus adaptive controllers can be divided into two basic types, direct and indirect methods (Astrom, 1986). Direct and indirect refers to the updated controller gains that are determined. A common representative of a direct adaptive controller is the model reference adaptive controller (MRAC). Here a model is used to tune the system gains to perform like the model. In general, the MRAC updates the controller parameters directly, (no intermediate calculations are required).

The self-tuning regulator (STR) is an indirect method, as a design procedure is used to determine controllers parameters. The STR uses model parameter estimation to determine current states for use in the real-time design calculations to adjust the controller parameters. It should be noted that there are similarities between the MRAC and STR techniques. The widely used explicit MRAC scheme of Landau (1979) for discrete SISO plants is based on the perfect cancellation of stable plant poles with controller zeros. The STR is the approach demonstrated in this text. The estimated grinding process model is used to update the controller design and subsequently select new gains, providing a fast and robust system.

It should be noted that the use of MRAC and other adaptive methods have been successfully applied in milling and turning. However, the limitations of these systems involve the speed the servo update of the complete system with a force loop. Indeed, servo update rates for adaptive control have been low. For example Kolarits (1990) had a servo update rate of only 9 Hz and Fussell (1988) had 8 Hz update, both for adaptive force control for end milling. Much of the process noise is still visible in these works especially in geometry transitions. This is probably attributable to the relatively slow force control update rates. New hardware speeds and the parallel processor designs of the control and estimation architecture permit a force control loop with a servo rate greater than 500 Hz. This faster force servo loop update rate yielding better force regulation, than those of previous efforts in milling and turning.

Process models have been successfully utilized in model reference adaptive control schemes in milling and turning, and as a mechanism for tool wear monitoring (*i.e.*, Tarn and Tomizuka 1989). Both fixed-gain and parameter adaptive controllers have been implemented to maintain a constant cutting force in milling and turning. Past research relied on slower computers, which resulted in a slow force control response. Closed-loop force control system sampling frequencies are generally low (frequencies of less than 3 Hz are not uncommon (Pein, 1992)).

As shown from control efforts in milling and turning, nonlinearities and parameter variations (most often from tool wear) have made the implementation of fixed-gain process controllers of limited value. Indeed, in many such cases instabilities or tool breakage occurred with fixed-gain process controllers. (Masory and Koren, 1980, 1985; Fussell and Srinivasan, 1988). Adaptive parameter estimation and process control are important aspects to increase process reliability and throughput using force control.

Much work, both theoretical and experimental, has been accomplished using model reference adaptive control in milling and turning operations (Landau, 1979; Stute, 1980; Tomizuka, 1984; Daneshmend and Pak, 1986; Fussell and Srinivasan, 1988; Ulsoy and Koren, 1993). These processes are single-point and a limited number of multi-point cutting edges, so the first research in adaptive machining process control began with these operations.

Digital, adaptive control theories have also been successfully applied in machine tool control using discrete models of the process and direct control design for stability (Lauderbaugh and Ulsoy, 1989). In many cases, the discrete models are based on simple first order data fits of sensor data such as a force to feed rate model in milling (Fussell and Srinivasan, 1988). Similar work has been accomplished in turning by Daneshmend and Pak (1986), where the process gain was estimated in real-time. Satisfactory performance was obtained for these systems even with simple models and controllers having slow sampling rates of 10 Hz. Nearly all of these digital, adaptive controller designs are based on pole-zero cancellation in some form. In his Ph.D. thesis, Kolarits applied adaptive pole placement for non-minimum phase systems, a form of the perfect tracking controller applying the inverse dynamics of the combined plant and process.

If the representation of a plant is accurate the perfect tracking controller is an excellent way to cancel unwanted dynamics using digital control, provided of course that the controller does not saturate the actuators in order to accomplish this. In cases where the inverse dynamics can cause an unstable pole in the controller, a zero phase error tracking controller (ZPETC) algorithm has been developed by Tomizuka (1987). In this scheme the stable plant zeros can be canceled, and the phase error of the unstable plant zero is canceled.

Grinding Control

A significant amount of work in off-line process optimization and parameter settings has been accomplished in grinding, including the collection of works by Malkin and Tonshoff. However, the research in real-time grinding process control has been limited. Sensors, models, and real-time estimation computational efficiency have been roadblocks to further advancement.

Nagao (1987) developed a flexible non-adaptive force controlled grinding system. The results obtained indicated that accuracy of such a system does not depend on the stiffness of the machine, which was low, but on the effectiveness of the force control.

Recent efforts of Kurfess and Whitney (1988, 1990), Ulrich (1992), and Elbestawi (1991) have focused on robotic disk grinding, which is somewhat different than the rigid wheel grinder addressed in this treatment. In these efforts, concentration was placed on the force control of a robot for deburring and not on the actual grinding process. Grinding was not estimated as a real-time process in these works.

As grinding is a multi-point, noisy, time-varying process having many parameters needed to define a grinding condition, only a limited amount adaptive control has been attempted. One of the first notable works in adaptive control in industrial grinding was that of Amitay, Malkin, and Koren, 1983. The objective of this research was the selection the optimal feedrates during grinding. In this approach, grinding power was the measured output variable for use in the adaptive controller. The gains of a PD feedrate controller were varied to maintain the maximum allowable power. The feedrate and the spindle speed were measured to determine the theoretical allowable power, based on the known material specific energy. This research was based on constant theoretical models without any real-time estimation. Force control was not directly addressed and only power was controlled. Since the power is a function of the tangential force and speed, any damage caused by the normal force variation was not addressed. Also the accuracy of power sensors used in this research and the ability of this approach to control force was not established.

The combination of a suitable grinding process model, a real-time estimator, and normal force control should provide improved force regulation (higher fidelity), permitting the machine tool deflections and material removal rates to be regulated more accurately, improving the repeatability of the resulting part shape (Ulrich, 1990). Ulrich showed that to maintain a specified metal removal rate in robotic grinding, the grinding normal force must be controlled.

In more closely related work, Tonshoff et al. (1986) conducted internal diameter grinding experiments with an adaptive force controller scheme. In this control scheme, the force variations were reduced when compared to non-adaptive PID force controllers. A third-order, discrete system model was developed, without a physical model basis. The identification of this recursive system model was determined in a separate grinding cycle and not in real-time during grinding. Therefore, the issue of changing process parameters was not addressed.

The adaptive control of grinding using an empirical, real-time grinding model has recently been addressed (Jenkins, 1996). This work was based on recursive estimation of the model parameters in Eq. (3.11) and multiple sensor input. The issue being addressed in this work is the development a real-time, adaptive model of the grinding process and the subsequent use of the identified model to control the grinding process, in terms of increasing the stability and fidelity of the normal force regulation. This desired normal force control improvement can be more explicitly stated as a force overshoot and variation reduction, while maintaining or improving the force bandwidth and the associated rise-time (or response time). This will provide a means for higher production rates without damage. Clearly by controlling the grinding normal force accurately, the basic grinding cycle can be shortened by removing spark-out or tarry time. Force and position control can be used to retract a grinding wheel after achieving a desired level of force and position.

3.6 Force Measurement

The measurement of force is a critical element in controlling the grinding process. This has generally been limited to either strain gage or piezo-electric type sensors. Strain gage based sensors exhibit no drift when loaded at constant force levels; however, their stiffness is generally low and dependent on the application and geometry. Although their sensor electronics generally have a high sampling rate, approximately 1,000 Hz sample rate, the natural frequencies of these sensors may limit their bandwidth and typically, their analog sensor signals are low pass filtered at a 100 to 200 Hz.

A piezoelectric-type (PZT) force transducer can be used to examine higher bandwidth force signals. Unfortunately, constant or quasi-static values of force are difficult to measure over long periods of time with this instrument there is charge “leakage” from the quartz crystal of the PZT that causes significant drift at constant load levels. The stiffness of the PZT dynamometer is substantially higher than that of the stainr gage type dynamometers with nominal values of approximately 10,000,000 pounds-per-inch (1.9 MN/mm).

3.7 Controller Design

A typical control application goal in grinding is the decrease of tarry time by the application of a stable force controller and providing greater fidelity for normal force regulation necessary in fine finishing. In order to perform the constant force grinding, a force controller must be designed for the grinding system. A block diagram of the complete grinding system and controller without the estimation and adaptive control elements is shown in Fig. 3.8. In this case the grinding system model consists of several functional blocks: (1) force controller, $G_{FC}(s)$; (2) PMAC (Programmable Multi-Axis Controller) position control loop, GPC(s); and (3) the grinding wheel to workpiece and process interaction, $G_{TWP}(s)$.

The PMAC position loop transfer function, $G_{PC}(s)$, is represented by block between the commanded position and the actual position of the axis, as measured by an encoder. Because the servo loop for position is very fast (<1 ms update) a linear, first order model of the position loop servo response, $G_{PC}(s)$, can be estimated using an auto-regressive, least squares parameter estimation technique. This is accomplished by performing several force servo experiments with the grinding wheel in contact with the workpiece. Note, it is important to obtain the position model at the anticipated force levels to determine

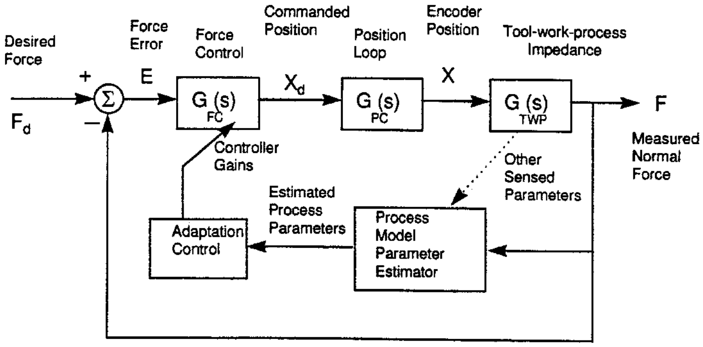


FIGURE 3.8 Plant and controller schematic for PMAC-based force control.

the correct position response, including the imposed force. As regressive models tend to favor higher frequencies, the samples are filtered to remove spectrum content at frequencies greater than desired, prior to identification.

The force control is accomplished by an outer force loop and an inner position loop (Fig. 3.8). The system model consists of several functional blocks: force controller, $G_{FC}(s)$, the PMAC position control loop, $G_{PC}(s)$, the tool-workpiece process interaction, $G_{TWP}(s)$, as well as the estimation and adaptation functions. The gains of the adaptive force controller are updated in real-time from the host computer that estimates the grinding process parameters and determines the force controller gains accordingly. Since force is being controlled via position, two control loops must be designed, a position loop and a force loop.

The servo controller for the position loop is usually accomplished with the resident PID-type (proportional, integral and derivative) servo algorithm of the PMAC. A PMAC can be used to perform multiple tasks in real-time. Motion programs are typically run to determine the position paths or to provide an outer control loop, as in this work. The PMAC system uses timer interrupts to control the microprocessor's action. Once every servo cycle (usually greater than 1 kHz), the timer interrupt is activated and the CPU (central processing unit) momentarily stops processing the motion program or other current task in order to update the axial positions (via the encoder inputs), sensor A/D (analog to digital) values, and perform controller functions. During this interrupt cycle the CPU determines the position errors for each axis and then uses the PID-type servo algorithm to update the control output voltage (DAC output) for each axis. Having completed this cycle, the interrupted task continues until completion or the next servo cycle.

The PC-based PMAC controller software interface provides a tuning utility to the determine the position loop gains for a desired bandwidth and damping ratio. These are based on the Ziegler-Nichols (Franklin, 1992) PID tuning techniques. This tuning utility is used to determine the PID algorithm gains for position control response during normal axis movement operations (while not engaged in force control). The specific Ziegler-Nichols tuning method used is the transient response method. In this method a step response of the open-loop system is obtained, defining values for the lag and the slope of the output, as shown in Fig. 3.9. The PID parameters are then calculated based on the desired closed-loop damping and desired bandwidth, assuming a second order response with a delay. Although this technique is based on continuous systems, for very fast sampling rates (greater than 20 times the desired bandwidth) the results also apply to discrete systems.

As the PMAC is a digital controller, there are many parameters and factors that affect the system response. A significant parameter is the velocity profile for commanded position moves, usually defined by an S-curve acceleration in the velocity and a constant acceleration section. Acceleration time may be limited by the settings for maximum permitted acceleration and velocity as well as amplifier saturation. To achieve the best performance there should be no saturation of the power, maximum acceleration, or maximum velocity. Thus, the PMAC settings for these parameters are adjusted on-line to achieve good (if not maximum) performance without saturation.

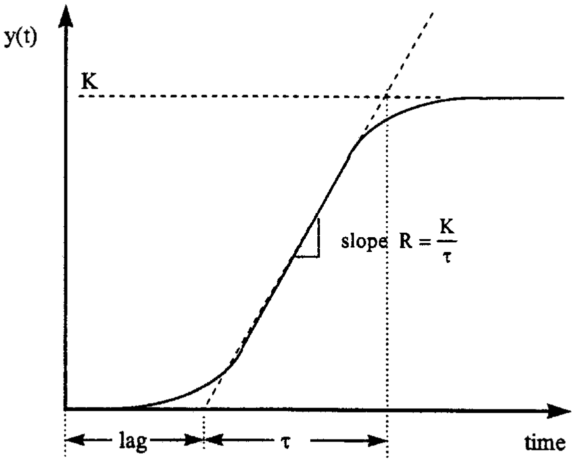


FIGURE 3.9 Ziegler-Nichols tuning.

Higher-frequency dynamics of the grinder structure should determined by modal analysis and measurement of the dynamic stiffness of the tool end of the servo stage. The primary structural resonances should be significantly beyond the position control bandwidth of the system. It should be noted that these structural vibrations are significantly higher than the significant frequency range of the process variation. Thus, the structural vibrations can be effectively separated from the process dynamics by digitally low-pass filtering. Specifically a second order (or higher) Butterworth low-pass filter should be implemented for anit-aliasing. The Butterworth filter is maximally flat in the pass band with respect to signal magnitude.

System Architecture: Parallel Processing and Implementation

A key element in implementation of adaptive control is the system architecture involving the host computer (typically a PC) and servo motion controller. Communication between the servo controllers and the host computer is accomplished by two means. The normal communication mode is via the PC bus. However, a faster mode of communication is available using the dual ported RAM. Implementation of parameter estimation routines and adaptive control algorithms performed in real-time using this dual-processing capability of the controller with the host computer will yield the best performance. The schematic of the computer process data flow is presented in Fig. 3.10. This configuration was implemented in Jenkins (1996).

The programmable multi-axis controller (PMAC) controls the servo control update, and performs the data collection and storage at a specified rate. PMAC stores the essential sensor data in special memory locations. This memory is addressable simultaneously by both the PMAC and host PC. Because of this dual access nature, this memory is called dual-ported RAM. Sensor and encoder data are stored in a rotary memory buffer, using the dual-ported RAM. The dual-ported RAM is common between the PMAC and the host PC, and may be accessed independently by either processor at any time. A pointer is provided to keep track of the last written memory location.

In this configuration the PC processes the information stored in the dual-ported RAM using a fixed length data window, much smaller than the total memory. A visualization of this windowing scheme implementation is depicted in Fig. 3.11. The windowing procedure used in the adaptive parameter estimation and control framework proceeds as follows. [Additional detail on the use of windowing techniques is provided in Danai and Ulsoy (1987).]

1. The first windowed sensor data are retrieved by the PC in a predefined window.
2. The PC program then performs all calculations for parameter estimation and determines the new gains during the second window.
3. The gain values are issued to the PMAC controller, and the cycle repeats.

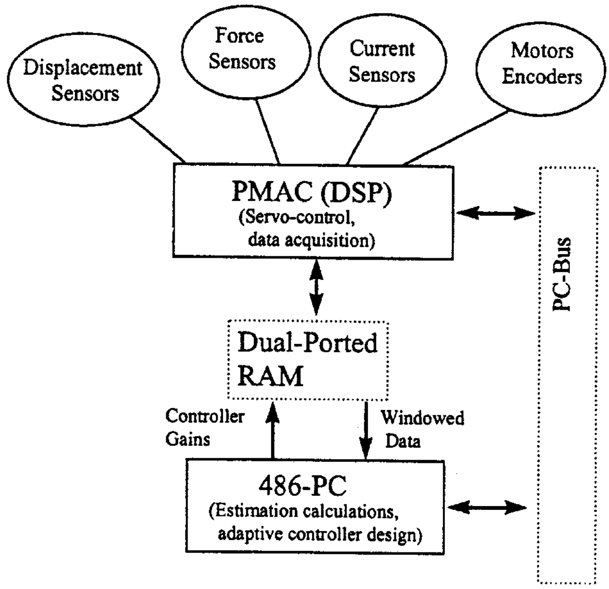


FIGURE 3.10 Computer information flow for implementation.

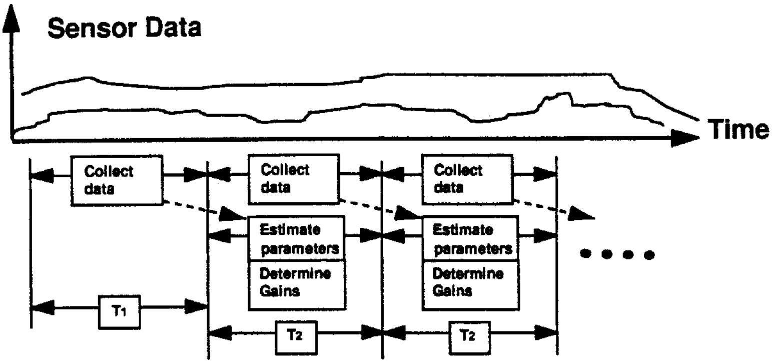


FIGURE 3.11 Windowing scheme for data collection and estimation.

Communication between the PC and the PMAC using the dual-ported RAM is the fastest mode possible, occurring within one PMAC servo cycle. As seen in the diagram above the PC can be used to determine new controller gains for the PMAC based on the latest sensor information and estimation calculations. The new controller gains are also passed transparently to the PMAC for use in the next servo cycle via the dual-ported RAM. This arrangement allows each CPU to perform its tasks independently, unhampered by the status of the other CPU. The PC performs calculations for estimation and control, while the PMAC provides a high rate servo control.

3.8 Adaptive Grinding Process Control

The grinding process has been modeled and has been shown to be a continuously, time-varying system (Ludwick et al., 1994). To better control the normal grinding force for the system, adaptive control methods are applied building on the successful real-time grinding model with on-line parameter estimation. The adaptive control strategy, process variation, and predictive power of the parametric model greatly influence the stability of the total system, and thus the degree of success.

An approach taken in adaptive force control effort for grinding is different from that of Tonshoff et al. (1986), discussed earlier. Tonshoff applied a fixed-gain controller to regulate the grinding force for an internal diameter grinder. In that work a deadbeat controller design is implemented for a third order recursive model between feedrate and normal force. Tonshoff used a separate off-line model estimation routine to determined the fixed-gains of the controller. In contrast, the work undertaken by Jenkins and Kurfess (1996) performs real-time estimation of the grinding process and uses a pole-zero cancellation approach to remove the process variation continuously in real-time.

Adaptive Controller Development

The development of an adaptive force controller for the grinding process can proceed directly from the plant model, Eq. (3.16), rewritten in Eq. (3.25).

$$G_{TPW}(s) = \frac{F_N(s)}{x_e(s)} = K_s\left(\frac{s}{s+\eta}\right) \tag{3.25}$$

where $\eta = K_P K_S\ V/A$. Combining Eqs. (3.11) and (3.12) yields the following continuous plant transfer function.

$$G_{PC}(s)G_{TWP}(s) = \frac{F_N(s)}{x_d(s)} = 203K_s\left(\frac{s}{s+\eta}\right)\frac{1}{(s+224)}. \tag{3.26}$$

From the earlier work, it has been shown that $\eta \ll 224$. Thus, variations in the process can dominate the system response. A fixed-gain controller can lead to poor performance as well as instabilities. Thus, an adaptive controller is desired to mitigate the effects of changing process dynamics and maintain a high MRR and a stable controller. From Eq. (3.26) it can be seen that the process dynamics can be eliminated by a pole-zero cancellation using a PI-type controller. The controlled system would then be of the form

$$G_{FC}(s)G_{PC}(s)G_{TWP}(s) = \frac{F_N(s)}{E_F(s)} = 203K_S\left(\frac{s}{s+\eta}\right)\frac{K_{\mathrm{PROP}}(s+\alpha)}{s(s+224)}. \tag{3.27}$$

In this scheme if the α (or K_I/K_{PROP}) of the PI controller in Eq. (3.27) is set to η, then the entire process grinding dynamics can be eliminated from the resulting equation. This reduces Eq. (3.27) to that of the previous situation of only a contact stiffness with a position loop, represented by Eq. (3.28).

$$G_{FC}(s)G_{PC}(s)G_{TWP}(s) = \frac{F_N(s)}{E_F(s)} = \frac{203K_S K_{\mathrm{PROP}}}{(s+224)} \tag{3.28}$$

where $\alpha = \eta$. Since a discrete equivalent system is needed for the sampled plant in order to implement such a scheme on the PMAC controller, a ZOH (zero order hold) equivalent discrete system is used.

$$G_{PC}G_{TWP}(z) = \frac{F_N(z)}{X_d(z)} = \left(\frac{z-1}{z}\right)Z\left(\mathcal{L}^{-1}\left[\frac{1}{s}203K_S\left(\frac{s}{s+\eta}\right)\frac{1}{(s+224)}\right]\right) \tag{3.29}$$

$$G_{PC}G_{TWP}(z) = \frac{203K_s}{224-\eta}\left(e^{-\eta T} - e^{-224T}\right)\left(\frac{z-1}{(z-e^{-\eta T})(z-e^{-224T})}\right) \tag{3.30}$$

denoting

$$
\begin{aligned}
C_0 &= e^{-224T} \\
f_1(\eta) &= e^{-\eta T} \\
f_2(\eta) &= \frac{203K_S}{224-\eta}(f_1(\eta) - C_0)
\end{aligned}
\tag{3.31}
$$

Substituting (3.31), into (3.30) yields

$$
G_{PC}G_{TWP}(z) = f_2(\eta)\left(\frac{z-1}{(z-f_1(\eta))(z-C_0)}\right). \tag{3.32}
$$

The discrete PI-type controller can be rewritten as follows:

$$
G_{FC}(Z) = \frac{K_{\text{PROP}}}{1+\alpha}\left(\frac{z-1/(1+\alpha)}{z-1}\right). \tag{3.33}
$$

Thus if η varies sufficiently slowly or the controller is sufficiently fast, α can be chosen such that

$$
\alpha = \frac{1}{f_1(\eta)} - 1 \tag{3.34}
$$

then complete open-loop discrete transfer function is defined as

$$
G_{FC}G_{PC}G_{TWP}(z) = \frac{K_{\text{PROP}}}{1+\alpha}f_2(\eta)\left(\frac{1}{z-C_0}\right) \tag{3.35}
$$

with an overall effective gain controller of

$$
K_{\text{EFFECTIVE}} = \frac{K_{\text{PROP}}}{1+\alpha}f_2(\eta) \tag{3.36}
$$

Equations (3.31) through (3.36) are used to determine the PI force controller gains in the controller design update of the self-tuning regulator. The effectiveness of the pole-zero cancellation is related to the process variation frequency spectrum and the gain update rate.

The implementation of this controller uses the windowed data handling scheme as previously discussed. Thus, the size of the data window, is the key parameter to the success of this approach. The smaller the window, the faster the gain update. However, as the sample window gets smaller, the variance of the parameter estimation increases. This reduces the confidence of the estimate and ultimately reduces the probability of an effective pole-zero cancellation.

Controller Stability and Bounds

An important issue in the successful application of adaptive control to nonlinear systems is the stability of the combined plant and controller. Several approaches have been developed by authors in this regard. The most common approach is to ignore the stability analysis, letting a successful implementation stand on its own merit without a rigorous stability analysis. This is dangerous and should be avoided.

The most notable stability analysis method, and the most popular employed, is assessing stability of a controlled system in the sense of Lyapunov (Slotine and Li, 1991). Lyapunov's direct method requires the existence of an energy-type, scalar function of the plant state that is positive definite, with continuous partial derivatives, and a negative, semi-definite time derivative. Several theorems are presented by Lyapunov for assessing local and global stability. The logic of this approach has been summarized by Slotine and Li (1991): "If this total energy of a system is continually dissipating, then the system must settle to an equilibrium point whether it is a linear or a nonlinear system." Finding a Lyapunov function can be difficult. It is mostly done by trial-and-error without any guarantee of success. This is a major limitation of this method of stability analysis. It should be noted that Lyapunov approaches can be especially difficult to apply to discrete systems.

Another approach for the controller design and assessment of stability is that of stochastic robustness (Stengel and Ryan, 1991). This work examines the stability of continuous linear time invariant (LTI) plants with uncertain parameters in terms of the probability of instability, via a stochastic root locus analysis. In this approach stability is not guaranteed, but the statistical probability of instability may be reduced. Guaranteed stability designs tend to be very conservative and slow. The results of this technique have been applied successfully in the aerospace field, with the space shuttle as a good example. The basis of the stochastic root locus approach is to develop a probability density function for stabile roots, using a Monte Carlo evaluation. (This method is applicable to Gaussian and non-Gaussian distributions.) The results of stochastic root locus can provide answers to the likelihood of closed-loop system instability within a desired confidence level. Although the method was developed for LTI continuous system, extension of the approach to linear time variant discrete system should be possible. Still, the most simple method to assess stability is to determine the bounds of estimated parameters that will cause instabilities and limit the parameter solutions to these bounds.

As the controller is stable by design for perfect estimation of the process parameters, the real plant stability is directly linked to the predictive power of the parameter estimation, the rate of parameter change, the controller update rate, and the update rate for the force controller gains. If the parameter prediction is perfectly accurate and the controller update is infinitely fast, then the stability is assured even without a Lyapunov analysis. Thus, the stability of the adaptively controlled-force grinding system is linked to the variation in the model parameters and the controller update rate.

A rigorous stability analysis in the sense of Lyapunov is complicated for STR systems, as the controllers are highly nonlinear. Indeed, the relationship of the regulated force to the estimated parameter can be quite convoluted and is difficult to attain in many instances. Mappings from the estimated grinding process parameters to the adaptive control parameters are necessary, and singularities must be addressed. Before any attempt is made to proceed down such a path, more obvious and tangible evaluations of stability can be performed. For the STR problem at hand, stability insight is gained by examining more basic approaches to the assessment of the system variation.

For the previous adaptive control example, the first approach in answering the question of the stability for the adaptive force controlled grinding system is a parametric examination of the effect of poor estimation and adaptive performance. The effect of poor performance can be seen by examining the combined open-loop, discrete plant and controller transfer functions given in Eqs. (3.30) and (3.35), combined to form

$$G_{FC}G_{PC}G_{TWP}(z) = \frac{K_{\mathrm{PROP}}f_2(\eta)}{1+\alpha}\left(\frac{z-\frac{1}{1+\alpha}}{(z-f_1(\eta))(z-C_0)}\right) \tag{3.37}$$

It can be seen that the poles of the open-loop system are all within the unit circle, so they are stable. The variable, α, is the ratio of the integral to the proportional gain and is always positive. Therefore the transfer function zero is always positive and less than or equal to one.

$$z = \frac{1}{1+\alpha} \tag{3.38}$$

Although an open-loop pole near unity is not desired in terms of the system performance specification, it is at worst marginally stable. Another possibility for instability is in the calculation of the proportional gain, given in Eq. (3.36). The overall gain is positive and is calculated in real-time to be stable dependent on the estimated grinding model. However, α is positive and determined by the controller. Thus even if the pole-zero cancellation is not accurate, the effects do not cause the system to go unstable; however, performance may suffer.

The implementation of this controller is based on the windowed data handling scheme of Fig. 3.11. Thus, the size of the window is the key parameter to the success of this approach. The smaller the window the faster the gain update. However, as the sample window gets smaller, the variance of the parameter estimation increases. This reduces the confidence of the estimate and ultimately the reduces the probability of an effective pole-zero cancellation.

The first approach in answering the question of the stability for the adaptive force controlled grinding system is a parametric examination of the effect of poor estimation and adaptive performance. The effect of poor performance can be seen by examining the combined open-loop, discrete plant, and controller transfer functions combined to form Eq. (3.39)

$$G_{FC}G_{PC}G_{TWP}(z) = \frac{K_{\mathrm{PROP}} f_2(\eta)}{1+\alpha}\left(\frac{z - \frac{1}{1+\alpha}}{(z - f_1(\eta))(z - C_o)}\right). \tag{3.39}$$

From Eqs. (3.39) and (3.33) it can be seen that the poles of the open-loop system are all within the unit circle, so they are clearly stable, even if only to a limited degree. Thus even if the pole-zero cancellation is not accurate the effects do not cause the system to go unstable; however, performance may suffer.

3.9 Summary

In this chapter a general discussion of the grinding process and applied control systems has been presented. A brief recent history, summary of grinding models, and adaptive control implementations have also been provided. An application of a state-of-the-art precision grinding adaptive control using a self-tuning regulator has been reviewed in detail.

Simplified modeling of the grinding process has been shown to be of value in real-time control. Several recursive techniques including a windowed Kalman filter and recursive least squares with a forgetting factor have been shown to yield excellent improvements in estimating the grinding model coefficients. Both these methods have a parameter related to the timeliness of the data. These are time weighting parameters for emphasizing the frequency content of the sensor data to be considered. Further examination of these parameters provides a basis for their selection, based on the best performance of the model.

The presented adaptive controller, based on a pole-zero cancellation of the estimated grinding process, revealed the use of parallel processors as a key element in the combined servo control and process estimation functions. The intrinsically stable, adaptive controller with estimator yielded a superior grinding force controller with higher fidelity force regulator with greater stability. This can be used to improve the surface profile following capability, as contrasted to a fixed-gain controller. This approach to grinding force control has demonstrated the potential for increasing productivity with the reduction of tarry time by use of the stable force control in plunge grinding, with higher fidelity in force regulation. The implementation of this approach in surface (traverse) grinding provides a superior surface following capability for fine finishing, as compared to fixed gain controllers.

Clearly more advancements in grinding control systems will be forthcoming as computer and control hardware become increasingly more powerful and accepted in industry to achieve greater competitiveness.

Nomenclature

ω = Rotational speed of the grinding wheel
λ = RLS forgetting factor ($0 \leqslant \lambda \leqslant 1$)
Θ = Parameter vector for RLS method
Ψ = Loss function
Ω = Angular velocity

Φ = Signal vector for RPS method
Φ, Γ = Discrete plant
Λ_W = Material removal coefficient
A = Area of the contact patch between the part and grinding wheel
a, a_c = Working engagement
a_P = Grinding contact width
B = Damping coefficient of the servo table
C = Number of active cutting points per unit area
d_e = Grinding wheel diameter
E, $e(t)$, ϵ = Error
e_i = Exponent
F_N = Normal force between the part and grinding wheel
$F_{N,C}$ = Normal grinding force from cutting action
$F_{N,R}$ = Normal grinding force from rubbing action
$F_{N,P}$ = Normal grinding force from plowing action
$\bar{F}_N$ = Mean normal force between the part and grinding wheel
$F_{T,C}$ = Tangential grinding force from cutting action
$F_{T,P}$ = Tangential grinding force from plowing action
$F_{T,R}$ = Tangential grinding force from rubbing action
F_{TH} = Threshold normal force between the part and grinding wheel
$G_{(q)}$ = Discrete plant model matrix
$H_{(q)}$ = Discrete error model matrix
h_{MAX} = Maximum chip thickness
k_1 = Coefficient
K_D = Derivative control coefficient
$K_d(s)$ = Dynamic stiffness
K_I = Integral control coefficient
K_o = Coefficient for material removal in grinding
K_P = Coefficient for material removal in grinding
K_{PROP} = Proportional control coefficient
K_S = Spring constant of the servo table
L = Workpiece thickness
l_C = Grinding contact length
L_K = Kalman Filter Gain
M_K = Time error covariance
P = Power
P_K = Measurement error covariance
Q = Material removal rate
q_a = Wheel wear by attrition
q_W = Wheel wear
r = Ratio of the theoretical chip width to height
$\bar{r}$ = Average radius of the grinding wheel
R_a = Reference signal, actual measurement
R_d = Reference signal, desired
r_g = Average grit aspect ratio
R_i = Inner radius of the grinding wheel
r_o, R_o = Outer radius of the grinding wheel
R_v = Covariance of white noise process v
R_w = Covariance of white noise process w
S_k = Sensitivity
u = Material specific energy

$u(t)$ = Control input
u_c = Material specific energy from cutting
u_P = Material specific energy from plowing
u_R = Material specific energy from rubbing
v,w = White noise processes
V_C = Cutting speed
V_f = Feed velocity of the grinding wheel
V_W, V = Speed of the cutting surface of the grinding wheel
W_{ij} = Neural network weighting function
x = Desired feed position of the grinding wheel
x_f = Actual feed position of the ground surface

References

Amitay, G., Malkin, S., and Koren, Y., Adaptive Control Optimization of Grinding, *ASME Journal of Engineering for Industry,* 103: 103, 1981.

Andrisani, D. and Gau, C., Multistage Linear Estimation Using Partitioning, *IEEE Transactions on Automatic Control,* AC-30: 2: 182, 1985.

Armstrong, B., Friction: Experimental Determination, Modeling and Compensation, *1988 IEEE Int. Conference on Robotics and Automation,* p. 1422.

Astrom, K. J. and Wittenmark, B., *Adaptive Control,* Addison-Wesley Reading MA, 1989.

Backer, W. R., Marshall, E. R., and Shaw, M. C., Forces in Dry Surface Grinding, *ASME Transactions,* 74: 51, 1952.

Bifano, T., Yi, Y., and Kahl, K., Fixed Abrasive grinding of CVD SiC Mirrors, *Precision Engineering,* 16: 2: 109, 1994.

Book, W. J. and Kwon, D. S., Contact Control for Advanced Application of Light Arms, *Journal of Intelligent and Robotic Systems,* 6: 121, 1992.

Brady, B., Trouble Shooting Diamond Wheel Failures, *Cutting Tool Engineering,* 37: 3: 32, 1985.

Brown, N., *Optical Fabrication,* MISC4476, Revision 1, Lawrence Livermore Laboratory, Livermore, CA, 1990.

Chou, K. C., Willsky, A. S., and Benveniste, A., Multiscale Recursive Estimation, Data Fusion, and Regularization, *IEEE Transactions on Automatic Control,* 39: 3: 464, 1994.

Chung, E. S., Chiou, Y. S., and Liang, S. Y., Tool Wear and Chatter Detection in Turning Via Time-Series Modeling and Frequency Band Averaging, *Manufacturing Science and Engineering, American Society of Mechanical Engineers, Production Engineering Division,* 64: 351, 1993.

Coes, L., *Abrasives,* Springer-Verlag, New York, 1971.

Colgan, J. Chin, H., Danai, K., and Hayashi, S. R., On-Line Tool Breakage Detection in Turning: A Multi-Sensor Method, *ASME Journal of Engineering for Industry,* 116:117, 1994.

Danai, K. and Ulsoy, A. G., A Dynamics State Model for On-Line Tool Wear Estimation in Turning, *ASME Journal of Engineering for Industry,* 109:396, 1987.

Danai, K. and Ulsoy, A. G., An Adaptive Observer for On-Line Tool Wear Estimation in Turning, Part I and Part II, *Mechanical Systems and Signal Processing,* 1:2:211, 1987.

Daneshmend, L. K. and Pak, H. A., Model Reference Adaptive Control of Feed Force in Turning, *ASME Journal of Dynamic Systems, Measurement and Control,* 108:215, 1986.

Drescher, J. D. and Dow, T. A., Tool Force Model Development for Diamond Turning, *Precision Engineering,* 12:1:29, 1990.

Elbestawi, M. A., Papazafiriou, and Du, R. X., In-process Monitoring of Tool Wear in Milling Using Cutting Force Signature, *International Journal of Machine Tools Manufacturing,* 31:1:55, 1991.

Elbestawi, M. A., Yuen, K. M., Srivastava, A. K., and Dai, H. Adaptive Force Control for Robotic Disk Grinding, *Annals of the CIRP,* 40:1:391, 1991.

El-Wardani, T. Sadek, M., and Younis, M. A., Theoretical Analysis of Grinding Chatter, *ASME Journal of Engineering for Industry,* 109:4:314, 1987.

Fawcett, S. C. and Dow, T. A., Development of a Model for Precision Contour Grinding of Brittle Materials, *Precision Engineering,* 13:4:270, 1991.

Franco, S., *Designing with Operational Amplifiers,* Addison-Wesley, Reading MA, 1988.

Franklin, G. F., Powell, J. D., and Workman, M. L., *Digital Control of Dynamic Systems,* Addison-Wesley, Reading, MA, Second Edition, 1990.

Fussell, B. K. and Srinivasan, K., Model Reference Adaptive Control of Force in End Milling Operations, *1988 American Control Conference,* 2:1189.

Hahn, R. S., Improving Performance with Force-Adaptive Grinding, *Manufacturing Systems,* 97:73,1986.

Hahn, R. S. and Lindsay, R. P., Principles of Grinding Part I Basic Relationships in Precision Machining, *Machinery,* p.5, July 1971.

Hahn, R. S. and Lindsay, R. P., Principles of Grinding Part II The Metal Removal Parameter, *Machinery,* 33: August 1971.

Jenkins, H. E., Kurfess, T. R., and Dorf, R. C., Design of a Robust Controller for a Grinding System, *The Third IEEE Conference on Control Applications,* 3:1579, 1994.

Jenkins, H. E., Process Estimation and Adaptive Control of a Grinding System, Ph.D. Thesis June 1996, School of Mechanical Engineering, Geogia Institute of Technology, Atlanta, GA, 1996.

Jeong, S. and Ro, P. I., Cutting Force-Based Feedback Control Scheme for Surface Finish Improvement in Diamond Turning, *American Control Conference,* 2:1981, 1993.

Kalker, J. J., Rolling with Slip and Spin in the Presence of Dry Friction, *Wear,* 9: 20, 1966.

Kalker, J. J., The Tangential Force Transmitted by Two Elastic Bodies Rolling Over Each Other with Pure Creep, *Wear,* 11:421, 1968.

Kolarits, F. M. Dynamic Modeling Of End Milling Forces and Surface Geometry and Adaptive Pole Placement Force Control, *Rensselaer Polytechnic Institute,* Ph.D. Thesis, 1990.

Konig, W. and Werner, G., Adaptive Control Optimization Concept, Technological Basics and Application, *Annals of the CIRP,* 23:1:101, 1974.

Koren, Y., Ko, T., and Ulsoy, A. G., Flank Wear Estimation Under Varying Cutting Conditions, *ASME Journal of Dynamic Systems, Measurement, and Control,* 113: 300, 1991.

Kubo, T., Anwar, G., and Tomizuka, M., Application of Nonlinear Friction Compensation to Arm Control, *1986 IEEE Int. Conf. on Robotics and Automation,* p. 722.

Kumar, K. V. and Shaw, M. C., The Role of Wheel-Work Deflection in Grinding Operations, *ASME Journal of Engineering for Industry,* 103:73, 1981.

Kurfess, T. R. and Nagurka, M. L., A Geometric Representation of Root Sensitivity, *ASME Journal of Dynamic Systems, Measurement, and Control,* 116:4:305, 1994.

Kurfess, T. R., Whitney, D. E., and Brown, M. L., Verification of a Dynamic Grinding Model, *ASME Journal of Dynamic Systems, Measurement and Control,* 110:403, 1988.

Landau, I. D., *Adaptive Control,* Dekker, New York, 1979.

Lange, F. and Hirzinger, G., Iterative Self-Improvement of Force Feedback Control in Contour Tracking, *IEEE International Conference on Robotics and Automation,* 2:1399, 1992.

Lauderbaugh, L. K. and Ulsoy, A. G., Dynamic Modeling for Control of the Milling Process, *ASME Journal of Engineering for Industry,* 110:367,1988.

Lauderbaugh, L. K. and Ulsoy, A. G., Model Reference Adaptive Force Control in Milling, *ASME Journal of Engineering for Industry,* 111:13, 1989.

Lee, J. H., Morari, M., and Garcia, C. E., State-Estimation-Based Model Predictive Control with On-Line Robustness Tuning Parameters, *1991 American Control Conference,* 3:2373.

Lewis, F. L., *Optimal Estimation,* John Wiley and Sons, New York, 2nd Ed., 1992.

Liang, S. Y. and Perry, S. A., In-Process Compensation For Milling Cutter Runout Via Chip Load Manipulation, *ASME Journal of Engineering for Industry,* 116:2:153, 1994.

Liang, S. Y., Lecture Notes *Machine Tool Analysis and Control,* ME-8403B, Georgia Institute of Technology, G. W. Woodruff School of Mechanical Engineering, 1994.

Ljung, L. and Soderstrom, T., *Theory and Practice of Recursive Identification,* MIT Press, Cambridge, MA., 1983.

Ljung, L., *System Identification: Theory for the User,* Prentice Hall, Englewood Cliffs, NJ, 1987.

Lo, C. and Koren, Y., Evaluation of Machine Tool Controllers, *1992 American Control Conference,* 1:370, 1992.

Ludwick, S. J., Jenkins, H. E., and Kurfess, T. R., Determination of a Dynamic Grinding Model, *1994 International Mechanical Engineering Congress and Exposition,* DSC-Vol. 55-2: 843, 1994.

Ma, Q. and Ro, P. I. Use of Force Feedback for Dynamics Error Correction in Diamond Turning For Surface Roughness Improvement, *ASPE 1994 Annual Meeting,* p.37, 1994.

Malkin, S. and Ritter, J. E., Grinding Mechanisms and Strength Degradation for Ceramics, *Journal of Engineering for Industry,* 111: 167, 1989.

Malkin, S. *Grinding Technology, Theory and Applications of Machining with Abrasives,* Ellis Horwood Limited, Chichester, U.K., 1989.

Marion, J. E., Strengthening of Solid-State Laser Materials, *5th IEEE/OSA Conference on Lasers and Electro-Optics,* p.232, 1985.

Marshall, E. R., and Shaw, M. C., The Size Effect in Metal Cutting, *ASME,* 74:61, 1952.

Masory, O. and Koren, Y., Adaptive Control Systems in Turning, *CIRP Annals,* 1980.

Masory, O. and Koren, Y., Real-Time Estimation of Cutting Process Parameters in Turning, *ASME Journal of Engineering for Industry,* 106: 400, 1985.

Maybeck, P. S. and Pogoda, D. L., Multiple Model Adaptive Controller For the STOL F-15 With Sensor/Actuator Failures, *IEEE Conference on Decision and Control, Including The Symposium on Adaptive Processes,* 2:1566, 1989.

Merchant, E. Mechanics of the Metal Cutting Process, *J. Appl. Phys.,* 16:207, 1945.

Morein, R. T. and Kalata, P., A Polynomial Algorithm for Noise Identification in Linear Systems, *5th IEEE International Symposium on Intelligent Control,* p.595, 1990.

Nagao, T. and Hatamura, Y. Development Of A Flexible Grinding System With Six-Axis Force Sensor For Curved Surfaces, *CIRP Annals*, 36:1:215, 1987.

Nagurka, M. L. Curve Performance of Rail Passenger Vehicles, *Ph.D. Thesis Dept. Mech. Eng. MIT,* Cambridge, May 1983.

Nunes, A. C., Rounded Approximate Step Function for Interpolation, *NASA Tech Briefs,* 17: 7: 84, July 1993.

Oaki, J., Stable Force Controller Design Based on Frequency Response Identification, *IEEE Workshop of Intelligent Robots and Systems,* p.1116, 1991.

Okazaki, Y., A Micro Positioning Tool Post Using a Piezoelectric Actuator for Diamond Turning Machines, *Journal of the Japan Society of Precision Engineering,* 54:7:1375, 1988.

Pappas, M. G. and Doss, J. E., Design of A Parallel Kalman Filter with Variable Forgetting Factors, *1988 American Control Conference,* p.2368.

Park, J. and Ulsoy, A. G., On-Line Flank Wear Estimation Using an Adaptive Observer and Computer Vision, Part I & II *ASME Journal of Engineering for Industry,* 115:30, 1993.

Pien, P. Y., and Tomizuka, M., Adaptive Force Control of Two Dimensional Milling, *1992 American Control Conference,* 1:399, 1992.

Purushothaman, S. and Srinivasa, Y., A Back-Propagation Algorithm Applied to Tool Wear Monitoring *International Journal of Machine Tools Manufacturing,* 34:5:625, 1994.

Rajmohan and Radhakrishnan, V., On the Possibility of Process Monitoring in Grinding by Spark Intensity Measurements, *ASME Journal of Engineering for Industry,* 116:124, 1994.

Rangwala, S., and Dornfield, D., Sensor Integration Using Neural Networks for Intelligent Tool Condition Monitoring, *ASME Journal of Engineering for Industry,* 112:218, 1990.

Reichenbach, G., Mayer, I., Kalpakcioglu, S., and Shaw, M., The Role of Chip Thickness in Grinding, *Transaction of the ASME,* 18:847, 1956.

Rowe, W. B., Inasaki, I., and Malkin, S., Applications of Artificial Intelligence in Grinding, *CIRP Annals,* 43:2:521, 1994.

Rowe, W. B., Qi, H. S. and Zheng, H. W., The Effect of Deformation on the Contact Area in Grinding, *CIRP Annals,* 42: 1: 409, 1993.

Rummelhart, D. and McClelland, J., *Parallel Distributed Processing, Vol. 1,* MIT Press, Cambridge, MA, 1986.

Salmon, S. C., *Modern Grinding Process Technology,* McGraw-Hill, New York, 1992.

Shaw, M. C., Fundamentals of Grinding Conference, Pittsburgh, PA, p.221, 1972.

Slotine, J. E. and Li, W., *Applied Nonlinear Control,* Prentice Hall, Englewood Cliffs, NJ, 1991.

Smith, M., Annaswamy, H., and Slocum, A., Adaptive Control Strategies for a Precision Machine Tool Axis, *Precision Engineering,* 17:3, 1995.

Stengel, R. F., and Ryan, L. E., Stochastic Robutsness of Linear Time-Invariant Control Systems, *IEEE Transactions on Automatic Control,* 36:1:82, 1991.

Stute, G. Adaptive Control, *Technology of Machine Tools,* Vol. 4, Machine Tool Controls, Lawrence Livermore Laboratory, Livermore, CA, 1980.

Szabo, B. and Wu, S. M., On-Line Techniques for Tool Condition Monitoring in Milling Operations, *IFAC Symposia Series - Proceedings of a Triennial World Congress,* 5:59, 1990.

Tarn, J. H. and Tomizuka, M. On-Line Monitoring of Tool and Cutting Conditions in Milling, *ASME Journal of Dynamic Systems, Measurement and Control,* 111:206, 1989.

Tomizuka, M., Zero Phase Error Tracking Algorithm for Digital Control, *ASME Journal of Dynamic Systems, Measurement and Control,* 109:65, 1987.

Tomizuka, M., Dornfield, D., Bian, X., and Cai, H., Experimental Evaluation of the Preview Servo Scheme for a Two-Axis Positioning System, *ASME Journal of Dynamic Systems, Measurement and Control,* 106:1, 1984.

Tomizuka, M., Oh, J. and Dornfield, D., Model Reference Adaptive Control of the Milling Process, *ASME Control of Manufacturing Processes and Robotic Systems,* p.55, 1983.

Tonshoff, H. K., Peters, J., Inasaki, I., and Paul, T., Modeling and Simulation of Grinding Processes, *CIRP Annals,* 41: 2: 677, 1992.

Tonshoff, H. K., Zinngrebe, M., and Kemmerling, M., Optimization of Internal Grinding by Microcomputer-Based Force Control, *Annals of the CIRP,* 35:1:293, 1986.

Tung, E. and Tomizuka, M., Feedforward Tracking Controller Design Based on the Identification of Low Frequency Dynamics, *ASME Journal of Dynamic Systems, Measurement and Control,* 115: 348, 1993.

Tung, E., Anwar, G., and Tomizuka, M., Low Velocity Friction Compensation and Feedforward Solution Based on Repetitive Control, *ASME Journal of Dynamic Systems, Measurement and Control,* 115: 279, 1993.

Ulrich, B., Liu, L., Elbestawi, M. A., and Srivastava, A. K., Force Control For Robotic Rigid Disk Grinding, *1990 American Control Conference,* 2: 1804, 1990.

Ulrich, B., Srivastava, A., and Elbestawi, M., Analysis of the Robotic Disc Grinding Process, *International Journal of Advanced Manufacturing Technology,* 7:2:82, 1992.

Ulsoy, A. G. Estimation of Time Varying Parameters in Discrete Time Dynamic Systems: A Tool Wear Estimation Example, *1989 American Control Conference,* 1: 68,

Ulsoy, A. G. and Koren, Y., Control of Machine Processes, *ASME Journal of Dynamic Systems, Measurement and Control,* 115:301, June 1993.

Ulsoy, A. G., Koren, Y., and Rasmussen, F., Control of Machine Processes, *ASME Journal of Dynamic Systems, Measurement and Control,* 105:107, June 1983.

Verkerk, J., The Real Contact Length in Cylindrical Grinding, *Annals of the CIRP,* 24:1:259, 1975.

Vermeulen, P. J. and Johnson, K. L., Contact of Non-Spherical Elastic Bodies Transmitting Tangential Forces, *J. Applied Mechanics,* p. 338, 1964.

Whitney, D. E., Edsall, A. C., Todtenkopf, A. B., Kurfess, T. R., and Tate, A. R., Development of an Automated Robotic Weld Bead Grinding System, *ASME Journal of Dynamic Systems, Measurement and Control,* 112:166, 1990.

Wiberg, D. M., Another Approach to On-Line Parameter Estimation, *1987 American Control Conference,* p. 418.

Wojcik, P. J., On The Design of an Adaptive Kalman Filter for On-Line Processing of Sensor Signals, *IEEE Conference on Decision and Control Including The 26th Symposium on Adaptive Processes,* p. 1605.

Xiao, G., Malkin, S., and Danai, K. Intelligent Control of Cylindrical Plunge Grinding, *1992 American Control Conference,* 1:391, 1992.

Ydstie, B. E., and Co, T., Recursive Estimation with Adaptive Divergence Control, *IEE Proceedings, Part D: Control Theory and Applications,* 132:3:124, 1985.

Younis, M., Sadek, M., and El-Wardani, T., New Approach to Development of a Grinding Force Model, *ASME Journal of Engineering for Industry,* 109:4:306, 1987.

4

Finite Element Modeling and CAD/CAM of Nett-Forming by Injection Forging

4.1 Injection Forging—Process and Component
Process Configurations • Component Forms
4.2 FEM and Applications in Metal-Forming
4.3 FE Modeling of Injection Forging
Modeling Requirement • Particular Modeling Technologies • Considerations for the Selection of FE Software • FE Modeling Procedures
4.4 The Applications of FE Modeling in Injection Forging—Examples
Analysis of Billet Stability • Definition of Limiting pdz Aspect Ratios • Preform Design for Injection Forging • Forming Sequence Design and Die-stress Analysis for Pressure-assisted Injection Forging of Thick-walled Tubes
4.5 Computer-Aided Design and Manufacture of Injection Forging
Methodology • Structure for Design and Assessment • System Development Considerations
4.6 Conclusions

Yi Qin
University of Strathclyde

Raj Balendra
University of Strathclyde

Finite Element approaches for analyzing nett-forming by Injection Forging are described with reference to modeling strategies, procedures, and applications; employment of Finite Element Modeling as an analysis engine to develop a design/analysis system for component/tool design for Injection Forging is also described with reference to design methodology, the structure of design-support, and software development considerations.

Nomenclature

l length of billet
d diameter of billet
L specific length of billet, l/d_0

$L1$	specific displacement of punch, l_i/d_0
v	displacement of the anvil
$L2$	specific displacement of anvil, v/d_0
t	thickness of flange or free-gap height of the die-cavity
w	tube thickness
T	specific flange thickness or pdz aspect ratio, t/d_0
r	exit radius of injection chamber
R	specific exit radius, r/d_0
D	specific flange diameter, d_1/d_0
$\bar{D}$	mean specific flange diameter,
σ	stress/strength
p	pressure
P	specific pressure (p/σ_y)
f	friction factor
μ	coefficient of friction
τ	friction stress

Subscripts

0	initial
1	current
a	applied
b	transmitted
n	normal contact
i	totally injected
Y	yield condition

4.1 Injection Forging—Process and Component

Injection forging is a process in which the work-material retained in an injection chamber is injected into a die-cavity in a form prescribed by the geometry of the exit (Fig. 1). The process is characterized by the combination of axial and radial flows of material to form the required component-form. In the 1960s, some interest was generated in injection upsetting [1]; it was developed with a view to extruding complex component-forms. The process configuration has since been the subject of research spanning fundamental analysis to the forming of specific components; branched components and gear-forms have been produced. The single-stage forming of such component-forms has been achieved by injection techniques; these forms were previously regarded as unformable by conventional processes. Currently, the nett-forming of some complex component-forms has been achieved by Injection Forging [2]. To date, several names have been used to describe this configuration—injection forming, injection upsetting, radial extrusion, side extrusion, transverse extrusion, lateral extrusion, and injection forging [2–22]. The name injection forging is used throughout this chapter.

Process Configurations

A number of injection forging configurations have been developed so far, such as injection against back pressure [3], injection with retractable anvil or injection-chamber [4], and pressure-assisted injection [5]. Other configurations are also possible; these may be defined with reference to the variation of injection-direction, billet geometry and forming constraints, tool-kinematics, and auxiliary media [2, 4, 6, 7]. With reference to work-material forms, injection forging configurations may be classified as three types—solid billets, tubular billets with supporting mandrels, and pressurizing media. These options are illustrated in Fig. 4.2.

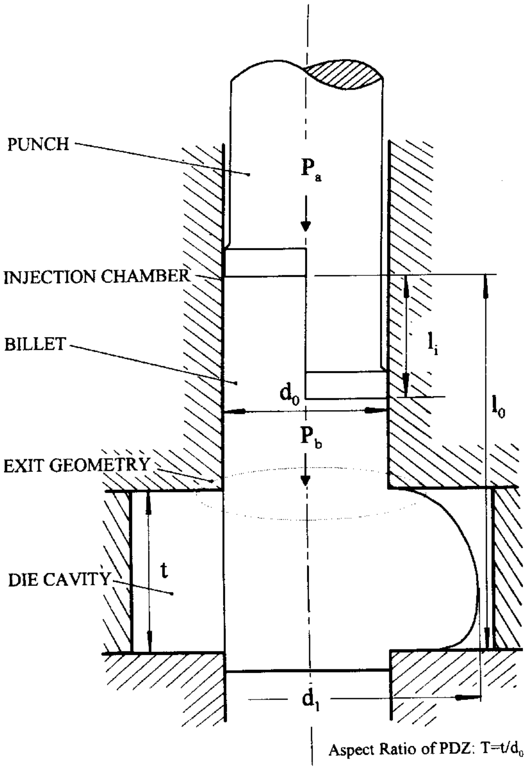
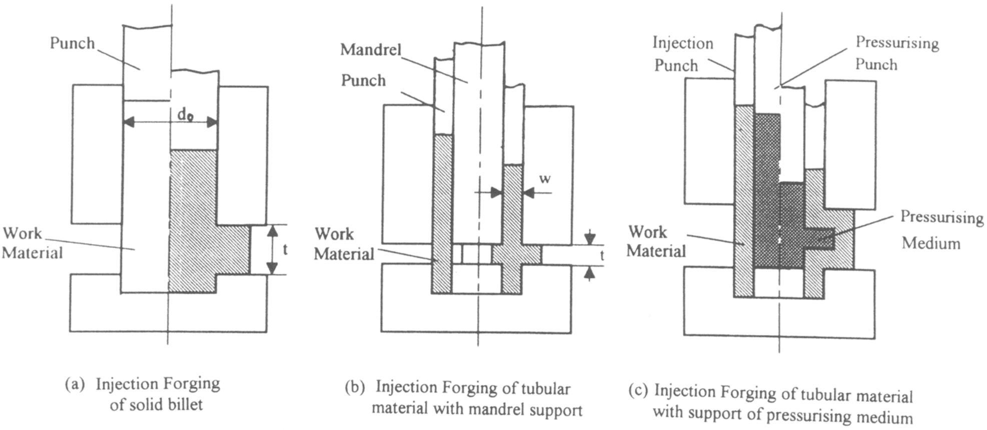


FIGURE 4.1 Simplified process model of injection forging.



(a) Injection Forging of solid billet

(b) Injection Forging of tubular material with mandrel support

(c) Injection Forging of tubular material with support of pressurising medium

FIGURE 4.2 Injection Forging configurations classified with reference to work-material forms.

Injection Forging of Solid Billet

Solid billets may be injection-forged to produce flanges, branches, or other geometric elements (Fig. 4.2a) [8–13]. The configuration with an active counter-pressure on the issuing flange, which can be achieved either by fluids or by matrix materials [3], may be used to prevent the initiation of fracture of the flange. The components can also be injection-forged by multi-axial extrusion [13] in which the billet contained in a closed die is extruded in several directions to produce a solid or hollow branched part in a single operation. Injection forging can also be combined with other forming processes; successful examples

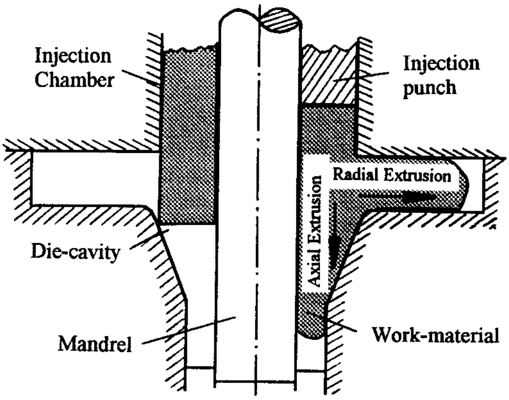


FIGURE 4.3 Combined radial and axial extrusion of tubular material.

include injection upsetting combined with back extrusion of a can, injection upsetting with subsequent forward can extrusion [14,15], and injection upsetting combined with back extrusion to produce a multi-branch flange component [16]. These combinations enable the reduction of intermediate forming operations, the number of tools and, hence, the cost of manufacture. In addition, the kinematics of tools can also be designed to adjust the forming forces and material deformation, e.g., using retractable injection chamber and anvil (or die) [4].

Injection Forging of Tubular Material with Mandrel Support

The tubular billets may be injection-forged to produce flanges, branches, and other component-forms [17] in which the inside diameter of the tube is invariably supported by a mandrel [7, 18–21] (Fig. 4.2b) to prevent collapse of the tube. The mandrel can either be used to support the entire length of the billet to prevent inward flow of the work-material, or it can support only the contained length in the injection chamber; the latter would allow the inward flow of material. The component can also be formed by the combined radial and axial extrusion of tubular material [19] to produce tubular flanged components with axially varied geometry (Fig. 4.3); the configuration was used to replace multi-stage forming by injection forging (a single operation). The ratio of material flow in the radial and axial directions, at any stage of the forming, is controlled by the die-cavity geometry and work-material/die-surface friction. The mandrels may also be replaced by retractable punches to iron the inside of the tube simultaneously with injection of the work-material [22] with a view to achieving control on the dimensions of the tube.

Injection Forging of Thick-Walled Tube with Assistance of Pressurizing Medium

Hollow components may be formed by injection forging in which a pressurizing medium is used to prevent collapse of the tube and to transmit the forming pressures to die-corners [23–27] (Fig. 4.2c). Pressurizing media can be either fluids or elastomers, such as polyurethane and natural rubber. The forming of thick-walled tubes usually requires the use of elastomers due to larger forming-pressure requirement than for the forming of thin-walled tubes. Since the bulk-flow of work-material is inevitable during the forming of thick-walled tubes, the forming sequence presents particular control issues.

Component Forms

In comparison with conventional forming, one of major advantages of injection forging is the ability to form complex component-forms, such as components with symmetric or asymmetric branches. The process is also able to achieve near-nett- or nett-shape definition of components [28]. It was shown that the manufacturing cost of an universal joint, when produced by injection forging, was two-thirds the cost when compared with conventional hot-forging [29].

Components formable by conventional cold forging are usually limited to axisymmetric-forms [30]; the shape of components in families varies only in the axial direction. Some specimens which were

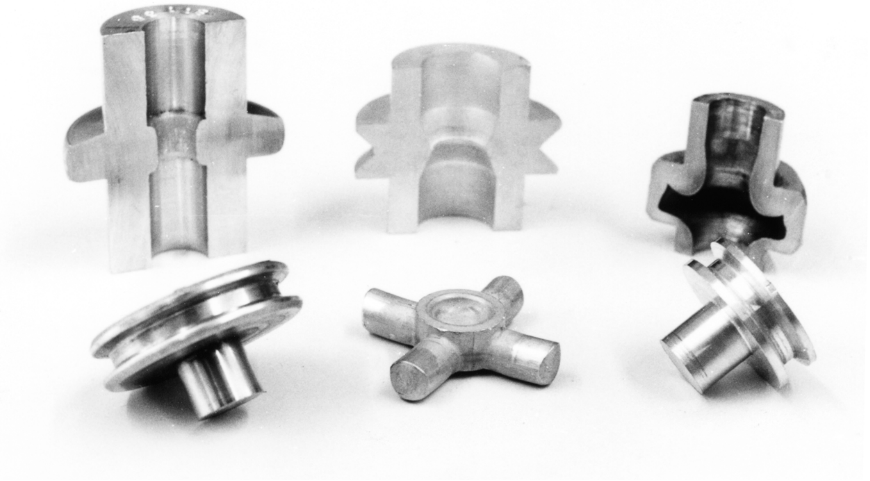


FIGURE 4.4 Examples of component-forms produced by injection forging.

difficult or impossible to produce by conventional cold forging were summarized in the classification [31]. Some of these have been formed by injection forging or by combination of processes, such as, spiders for universal joints [28, 29], the bodies for diesel injectors [32], pipe flanges [19], solid flanges with square cross-sections [33], alternator pole parts with claws [16], and staggered branches [34]. Some of the components formed in Manufacturing Engineering Research Center (MERC) of the University of Strathclyde are shown in Fig. 4.4.

Injection forging may be used to form components with radial variations in geometry. Generally, it is possible to produce components with flanges, stepped shafts, and secondary (local) shape elements; the shape elements can be solid or tubular, cylindrical or square cross-section, and can be incorporated throughout the component [2]. In addition, components which may be formed by injection forging can also contain axial variations. The combination of radial extrusion with forward and backward extrusion has enabled the forming of such components in single stage [14-16, 19]. Some component-forms, which had been classified with reference to axial variations of component-geometry for forming by conventional cold forgings [30], can be formed more effectively by injection forging; nevertheless, complex tooling is required to effect such forming. With reference to all possible variations in component-forms, components which may be formed by injection forging may be classified by the approach depicted in Fig. 4.5.

4.2 FEM and Applications in Metal-Forming

Design, with a view to forming flawless components which satisfy the specified performance and quality requirements, is a universal aim. To achieve this quality of design, the design practitioner has to have a clear appreciation of material deformation and tool behavior during forming. The analysis and design of metal forming has been affected by the use of several process-modeling methods—analytical, numerical, and experimental methods. Commonly used analytical methods are Slab, the Slip-line field, and Upper-bound methods; numerical approaches include finite element and boundary element techniques. Experimental methods include Physical Modeling and visioplasticity; the latter may be classified as a "half-analytical" approach since it combines experimental and analytical techniques.

Finite-element method (FEM) was originally developed as a technique for the analysis of structures; a number of structural elements may be analyzed as an assembly. The feasibility of applying this technique to analyze complex structures was enhanced by the development of digital computers. The method was subsequently extended to solve problems relating to the deformation of continua which may be subjected

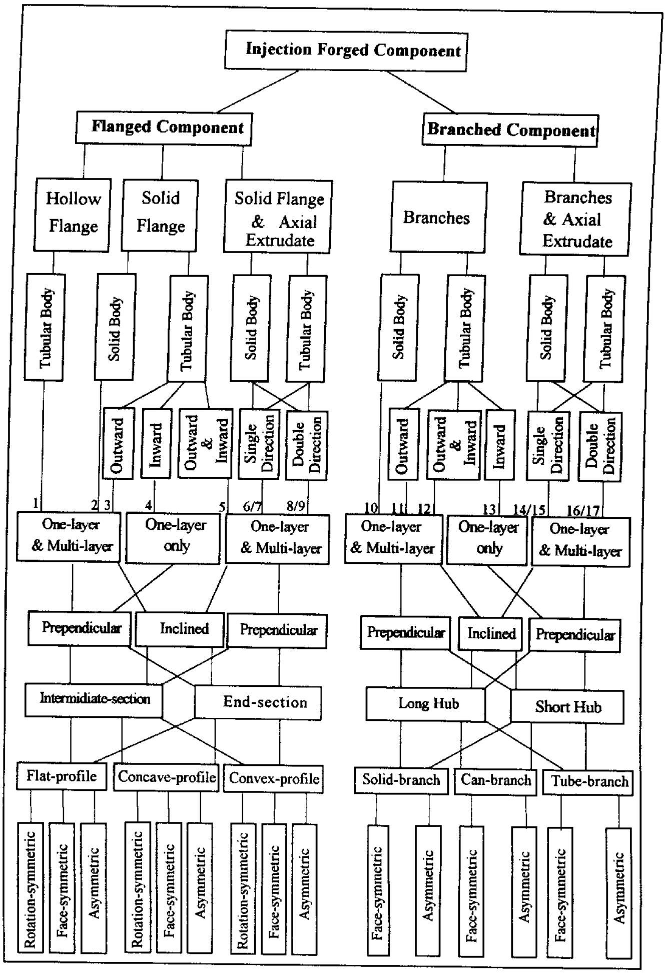


FIGURE 4.5 Classification of component-forms formable by injection forging.

to discretization into a number of small geometric elements (similar to individual structural elements). Progress in developing plasticity-theories and numerical methods for solving finite-element equations enabled extending the use of FE to the analysis of the plastic deformation of engineering materials. The fundamentals of FE analysis of material deformation has been dealt with extensively [35, 36].

A major advantage of FEM is its capability of obtaining detailed solutions of the mechanics in a deformed body, namely, velocities, shapes, strain, stress, temperature, and contact pressure [35]. The deformed body can be of complex geometry which would be difficult to subject to analytical methods. In fact, only since FE was introduced to the analysis of material deformation has it been possible to quantify the influence of all the major parameters on the flow of material.

The finite-element method was originally applied to simple metal-forming processes, such as upsetting and forward extrusion, for which the rigid-plastic formulation was popular. The rigid-plastic formulation (or flow formulation) is easy to implement in programming and is of high computational efficiency; it is particularly suitable for implementation on small computers. These reasons contributed to its wide use in the 1970s and 1980s. The application of the rigid-plastic formulation was gradually extended from plain-strain models and axisymmetric models to, recently, 3D models [37]. Further, with progress in large-deformation theory and appropriate numerical techniques, the elastic-plastic formulation has been gradually applied to metal forming since the 1980s, particularly in cases where the elastic components had to be considered in the analysis. The elastic-plastic formulation is able to simulate more realistic material-flow and, hence, to produce more accurate deformation and stress results. The method, however, requires more complex numerical treatment.

Early applications of FE in metal forming were mainly for the simulation of the sequence of die-filling and the computation of tool-stresses; to enable this, the forming process and work-material property was somewhat simplified. Due to the theoretical breakthrough in plastic formulations and numerical solution methods, rate-dependence and temperature effects could be incorporated into analyses as well as the solution of thermally-coupled problems. FE simulation is currently capable of being applied to warm and hot forming processes [38, 39]. The dynamic characteristics of forming-processes have now been evaluated, particularly since the advent of "EXPLICIT" FE methods. Success in the application of FE simulation for analyzing metal forming was also due to the significant progress which had been made in numerical modeling of contact and friction problems [40], such as large sliding 3D contact problems [41]. The feasibility and robustness of simulation of material sliding along the tool-surfaces is essential for the simulation of large-deformation forming processes.

Early development in the applications of FE for analyzing metal forming focused on small-scale software which was usually programmed for a particular forming process or a specific component-form. These are gradually being replaced by the development of large-scale, commercial plastic-analysis software, such as ABAQUS, MARC, Forge3, and DEFORM. The application of this software was further promoted by the availability of efficient pre- and post-processing software, such as PATRAN, SDRC I-deas, and ABACAE. More recently, adaptive and automatic remeshing techniques have been added to some software (e.g. MARC, DEFORM, and Forge3), which added further flexibility and efficiency to FE simulation of metal-forming processes. Taking FE software as a neutral engine, some CAD/CAM or CAE systems are being built [42] with built-in artificial intelligent techniques for the development of Knowledge-based Systems for design of forming operations [43–45].

Currently, FE simulation is being applied to various fields of analysis and design of metal forming. These may be summarized as follows:

General flow simulation and deformation analysis—The bulk flow of work-material may be simulated using FE to analyze the property of deformed material, e.g., stress-strain condition in the work-material, and to determine forming force requirement and pressure contour on the tool-surface. The simulation has been applied to the analysis of various bulk-metal forming processes: upsetting, extrusion, open- and closed-die forging, and shape rolling. Current efforts in developing FE approaches concentrate on the analyses of more complex process conditions or the forming of new materials. FE is also widely used to analyze sheet deformation in bending, blanking, stamping, and deep drawing. EXPLICIT FE methods are particularly efficient for these applications. FE is often used to simulate the spring-back, wrinkling, and fracturing of the sheet.

Product-quality control—A forming process may be optimized using FE analysis with reference to the product-quality. This may be achieved by the prediction of the development of work-material flaws, e.g., flow imperfections (buckling, folding, underfilling) and the fracture of material. For the latter, the criteria of ductile fracture of work-material is usually incorporated into FE analysis to predict fracture. Process parameters, preform geometry, and forming sequence may be designed to enable "optimal" pattern of material flow or "optimal" state of stresses and strains in work-material, with a view to preventing the development of the flaws. Forward simulations with trials or backward tracing simulation may be used to achieve these. The optimization algorithm may also be incorporated into FE formulation to establish "automatic" FE-optimization procedures, such as penalty finite element methods and shape optimization methods. These methods are currently limited to the analysis of simple forming operations. Further effort is required to improve the efficiency of the methods.

Friction plays an important rule in prescribing the quality of product. A variety of friction models are currently available, each of these being limited to a particular range of pressure, temperature, and type of interfacial material. Friction models may be chosen with reference to the accuracy of FE results and the numerical stability of contact analysis. FE formulation also provides efficient means for friction design by which complex descriptions of friction can be efficiently incorporated into the FE model.

The analysis may be conducted to achieve “optimal” friction with reference to specific requirements, such as stability of the workpiece, property of the deformed material, component-form error, and forming force requirements.

Tool-defect prediction and life analysis—The reduction of tool wear and the prevention of fracture of tools are two major aspects which are invariably considered for improving tool life. Prediction of the pressure on the tool-surface can be determined by FE analysis; reduction of the amplitude of the pressure may be enabled or its distribution may be optimized by optimizing the geometry of the tool to improve the flow of work-material. FE analysis is also able to provide detailed information of stresses and strains in the tool; the information may be used to initiate fracture mechanics analyses to predict the development of the cracks. For example, by evaluating the stresses and strains around a crack-tip, the fracture mechanical values such as K-factor or J-integral can be calculated and the yielding crack growth rate and the crack direction can be predicted. Based on these, die working-surface or tooling construction may be optimized with respect to the life considerations.

Forming-machine/tooling system analysis—A forming process may be regarded as operations conducted in a system which includes several elements such as machine, tooling, workpiece, and auxiliary media. Static and dynamic behavior of the machine would influence the behavior of tooling and consequently influence the component-form, such as influencing the concentricity of component and dimensional error. A forming operation may be influenced by tool, particularly by parameters such as die-cavity-geometry, tool material property, and tool surface coating and wear. The accurate prediction of tool-life and component-form error can only be achieved with reference to the deflection of machine, tool, and work-material. While this would be difficult to achieve by using any analytical method, FEM is an efficient tool of the modeling of the forming system; FE modeling of machine, tool and work-material as integrated models is currently time-consuming, but promises scope for more complete analyses of forming processes.

Although evidence indicates that FEM is an efficient tool for analysis and design of metal forming operations, the applications of FE have not been as popular and as efficient as expected. It has been proven that both efficient application of commercial FE codes and programming of a new FE software requires a high level of expertise. Although effort has been expended to improve the use of the FE software by continuously adding updated pre- and post-processing functions to the software, FE codes may not be treated as a “black box.” Computational failures could occur at any stage of FE modeling and computation. The successful modeling of a complex forming-process relies, to a large extent, on previous experience and universal rules may not be derived to enable easy access for the beginner. Further, modeling technologies are required for the simulation of particular forming processes and process-conditions, particularly the simulation of the dynamic characteristics of development of flaws. Innovative modeling techniques, based on available FE software, may resolve some of the current difficulties while others may require the further development of FE-theory and the method of numerical-solution.

4.3 FE Modeling of Injection Forging

Modeling Requirement

Early attempts to develop a predictive model of material flow in injection upsetting relied on upper-bound techniques [46]. The model was based on the following assumptions: the material flow within the die-cavity was in a steady-state throughout injection, the transmission of punch pressure along the injection chamber was effected with insignificant friction losses and the dimension of the upsetting cavity in the thickness direction was treated as a continuous variable. Another attempt at modeling treated the flange as an expanding thick-walled tube, the plastic deformation of which provided a definition of the energy requirement [1]. However, the hardness test of a formed flange by injection upsetting [47] showed that the strain-field was not as uniform as would have been the case for the expansion of a thick-walled tube; the forming of the flange may not be assumed to result from the uniform radial flow of material.

An upper-bound analysis was also conducted to define the force requirement to injection upset against hydrostatic pressure [48]. The work extended previous formulations [46] in two aspects—calculation of friction losses along the tool-workpiece interfaces, and assessment of the effect of internal surface discontinuities with reference to a variety of geometrical shapes. In this analysis, friction at billet/injection-chamber interface was described using a constant friction formulation and was assumed to occur at all work-material interfaces. The thinning of the flange was also investigated, from which it was concluded that this phenomenon could account for some of the discrepancy between upper-bound models and experimental data.

Other applications of upper-bound techniques include the prediction of extrusion pressure for the forming of tubular components and multi-branched solid components. A comparison of a simple parallel velocity-field with a triangular velocity-field for upper-bound analysis was also conducted for injection upsetting [49] from which it was concluded that the simple parallel velocity-field provided a better upper-bound value of injection pressure than the triangular velocity-field. The upper-bound technique was also used for predicting the load for forming a component with rotational-asymmetry as is the case of segmented flanges (3D component) [33]. In this analysis, the effect of the segment angle on the load characteristic was analyzed. The analysis gave higher load values than experimental results and this over-estimate was 25% at the initial stage and 10% at final stage of injection.

The capability of process modeling using analytical approaches was assessed by referring to different process conditions [50]. The results showed that previous upper-bound models over-estimated the punch pressure to initiate injection and under-estimated that to produce the flange. The research also indicated that the complexity of process parameters prevented the development of comprehensive models [51], for example, the exit-geometry has a significant influence on the forming-pressure in injection forging. It is, however, difficult to consider this parameter using analytical methods. To date, almost all upper-bound analyses of injection upsetting disregarded geometric transitions in the die-configuration. In fact, experimental evidence [47] suggests that a unique upper-bound field cannot cope with the entire range of flange thicknesses and the distribution of shear surfaces is dependent on the extent to which the flange has been formed. In addition, although some upper-bound analyses had been conducted by including friction in the analyses, this was attended to using only the constant friction law. The calculation of equivalent friction-coefficient from experimental data [51] showed that the friction in the injection chamber would not be constant and increased with the interfacial pressure. Analytical approaches are also unable to predict the development of flow-related flaws, such as the lateral sliding of the billet and the folding of the free-surface of the work-material [5, 11, 12, 52].

Injection forging, both for solid and tubular material, is characterized by the injection of work-material, which is contained in the injection-chamber, into a die-cavity or an annular space. The process may be segmented into these components, each of which retains a quantifiable character: the injection chamber, exit-geometry, and die-cavity (refer to Figs. 1 and 2). The modeling of the process is required with reference to characteristics of work-material deformation and forming pressure in these components. There would be no significant bulk-flow in the injection chamber; the pressure in the chamber would, however, be several orders higher than the yield strength of the work-material. Process modeling may be applied to the range of high pressures encountered in the chamber. Accurate prediction of the pressure distribution in the injection chamber would require the use of friction models which are pressure-dependent. Proposed models would have to be applicable to the analysis of the influence of exit-geometry on material flow. The flow of work-material in the die-cavity largely depends on die-cavity geometry. Analytical approaches can, however, only be capable of considering simple die-cavity forms. The requirement to achieve the nett-form definition of complex component-forms by injection forging suggests that the forming operations should be designed to achieve flawless components with the required level of performance and characteristics. The capability, such as simulating the development of material-flow flaws, quantifying the property of deformed work-material, and analyzing component-form errors due to tool- and work-material-elasticity, is an essential requirement for the modeling of nett-forming by injection forging. FEM is currently the most efficient tool for achieving these.

Particular Modeling Technologies

Previous research in injection forging was largely experimental or dependent upon the use of simplified analytical methods; FE simulation was only used in a few analyses. A preliminary attempt to use FE to analyze injection forging was with a view to predicting the punch pressure for the injection of a flange with a flange-compensation pressure [53]. Using a rigid plastic finite element code (PLADAN), combined radial and forward extrusion of a can was simulated [15]; material flow patterns and tool stress were analyzed for small aspect ratios of pdz (T, refer to Fig. 4.1). The research did not progress to the simulation of the development of flow-dependent flaws. FE simulation was used to analyze material-flow during injection forging of tubular material with mandrels [54]. Comparison between FE-predicted material-flow patterns and that observed from experimental specimens showed differences which may have resulted from the indiscriminate use of commercial FE software.

Only recently has large-scale FE simulation been applied to the design and analysis of injection forging for both solid billets and tubular materials [5, 11, 12, 20, 21, 25–27, 52]. The approaches and techniques have been developed to analyze the pressure losses in the injection chamber, the stability of the billet, flow characteristics of material in the die-cavity, and to design preforms and the forming sequence to prevent the development of flaws.

Modeling of Friction in Injection Chamber

The volume of material which can be injected into the die-cavity is dependent upon the friction characteristics of the injection chamber. These characteristics also have a significant influence on tool-stress. The research revealed that the pressure losses in injection chamber (pa-pb in Fig. 4.1) for a long billet could be substantial—40 to 60% of the injection pressure being expended to overcome friction in the injection-chamber [9].

The modeling of friction in the injection-chamber is of fundamental and practical interest, and may be effected by incorporating an appropriate friction model into FE analysis. There are two aspects which have to be considered for friction modeling of a forming operation. First is the accuracy of the friction model in replicating the prevalent interfacial friction conditions. Second is the numerical convergence when incorporating a friction model into FE simulation. Currently, three friction models are popular for the analyses of cold forging: Amonton's Law (Coulomb friction), Constant Friction Law, and interfacial pressure-dependent law (Bay and Wanheim's model [55]). The first two assumed either a constant friction coefficient or a constant friction factor, while the latter considered real contact between work-material and tool-surface (rough material-surface is considered).

The comparison between computed punch pressures and experimental results suggested that all three friction models were unable to accurately predict the friction characteristics at the injection-chamber/billet interface for high interfacial pressures. It was recognized that friction characteristics at the injection-chamber/billet interface would have to be defined with reference to the interfacial pressure which is much higher than these encountered in the application of three friction models. Since the existing approaches for measuring friction and those for determining friction by measuring the deformation of material (such as ring compression test or twist test) are limited to low working pressures, it is difficult to define friction characteristics under high pressure ($p/\sigma_y > 5$) levels using experimental approaches. The measurement of friction would have to be conducted with conditions which match that prevailing in the process under investigation. Injection forging is performed under high injection pressure; for a low alloy material, the pressure in injection chamber could be 10 times the yield strength of the work-material. It would be difficult to achieve this level of pressure using conventional friction tests. The measurement of friction in the injection-chamber would, therefore, have to be based on the measurement of the average pressure-loss in the injection chamber. As an approximate, equivalent friction coefficient, which is pressure-dependent, was determined from the measurement of pressure loss in the injection-chamber. These values were imported to an interface module of an FE software to define pressure-dependent friction characteristics, which were to be used in the contact analysis for the simulation of injection forging.

Simulation of Buckling During Injection Forging

Several types of flaws develop during the injection forging of components; among these, a prominent form results from the instability of the unsupported length of the billet. For injection forging without an end-constraint, the material in the die-cavity may buckle or slide laterally along the anvil when the unsupported length exceeds a threshold value. Subsequent die-filling is effected by asymmetrical deformation of the billet, resulting in components with folds or other flaws. The ability to simulate unstable deformation of the billet is essential for the definition of an aspect of formability in injection forging.

Slab and Upper-Bound methods are only applicable for the calculation of force or pressure characteristics, but are not designed to predict the development of flaws. Euler's formula for struts may be used for prediction of elastic buckling of long columns (height/diameter ratio $L = 3$). The extended formula for the plastic buckling analysis of columns is generally inapplicable for prediction of buckling in material-deformation processes; FE analysis may be applied to consider such conditions.

In general FE simulation, the work-material was invariably assumed to be perfectly uniform—geometrical and metallurgical defects are assumed not to prevail. The simulation of post-buckling behavior of the billet may not be possible if the model is perfectly symmetrical in terms of in the geometry, material-property, and loading. Asymmetry in billet geometry and inhomogeneity in metallurgical structure will prevail in engineering materials, though accurate measures of these would be difficult to define. The prevailing factors which would influence the stability of the billet in injection forging include the clearance between the billet and the injection chamber, exit-geometry, billet geometry, work-material property, die-cavity geometry, and the friction at billet/anvil (or die) interface. The strategy of assuming imperfections in FE models has been used for the simulation of buckling behavior [11]. The following approaches are regarded as feasible for simulation: the material may be assumed to be either loaded eccentrically, or prescribed a geometric and material-property imperfection. The degree of imperfection can be defined with reference to test results. In the injection upsetting model, the scope for eccentrical loading is relatively low as the injection-chamber provides an exact alignment between the punch and the billet; consequently, only geometric eccentricity of the billet in the die-cavity may be considered. Material inhomogeneity may also be defined by prescribing different flow-strength values to strands of elements in the model.

Geometric eccentricity of the billet may be modeled with reference to the clearance, between the billet and the injection chamber, which is specified for the process. It is difficult to quantify accurately the eccentricity that prevails for a batch of tests. The value which is to be used for the simulation is, therefore, determined by comparing simulated and experimental results. Several iterations are usually necessary to match material deformation-modes to FE models. A further consideration is that the length of the billet should be sufficient for deformation to continue beyond buckling. In order to avoid the difficulty of defining the effective aspect ratio when a large exit-radius is incorporated in the injection chamber, a small radius is used in the simulation. However, small exit-radius would result in difficulties for the simulation of material flow since small elements could have be used to overcome problems associated with the transition of elements around a small radius.

In the cases where billet buckling is significant, the simulation should incorporate particular numerical treatments, such as "RIKS" procedure in ABAQUS [56], to cope with numerical instability that accompanies the buckling of billets. Specific values of equilibrium tolerance and contact iteration parameters would have to be selected in anticipation of the rapid transition that occurs at the initiation of buckling. These specific values are chosen to prevent "hunting" between adjacent contact elements of opposing signs immediately prior to the incidence of buckling; failure to attend to this requirement would result in an inability to predict the post-buckling behavior of the billet since the simulation would not converge. An example of FE-simulated deformations for unstable conditions is shown in Fig. 4.6.

Simulation of the Folding of Material During Injection Forging

During injection forging, a flaw may develop due to the folding of free surfaces. The simulation of the development of folds is of significance for understanding the mechanism of the flaw development; the approach may be developed to eliminate flaws. The simulation of the development of the folds depends on the ability to simulate the flow of material on itself (contact between free-surfaces); previous approaches

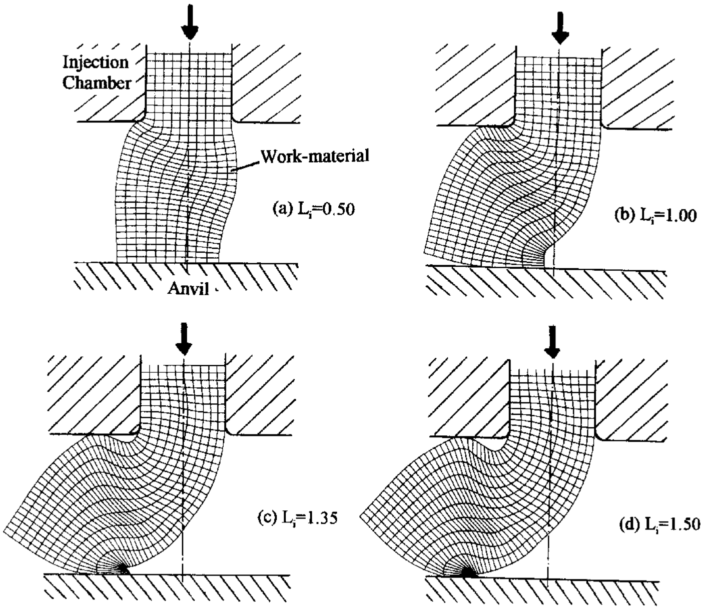
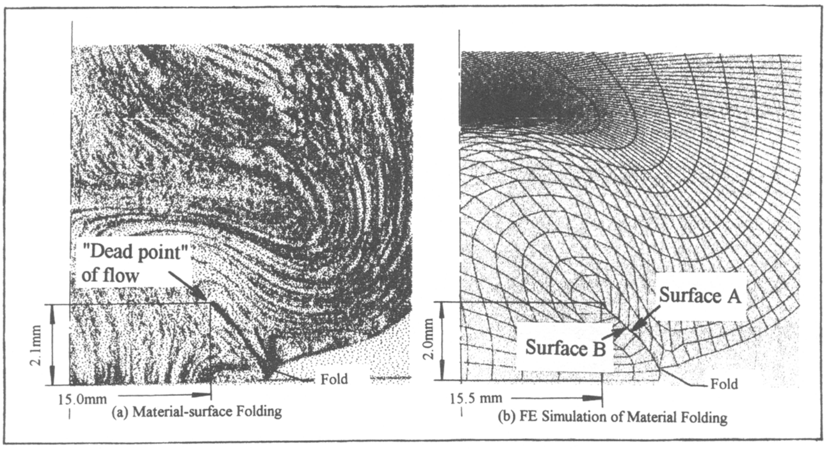


FIGURE 4.6 Illustration of unstable deformation of a billet ($T = 1.64$, $\mu = 0.01$)



(a) Material-surface Folding

(b) FE Simulation of Material Folding

FIGURE 4.7 Simulation and experimental comparison of a fold.

did not enable such simulation. The simulation had to be terminated when folding was initiated or was continued by treating folds as continua. In order to simulate the development of folds, software that has functions of simulating sliding between two deformed bodies is required. ABAQUS [56] retains a type of sliding element, which can be used to simulate the flow of the material on itself.

By observing material flow, a "dead point" may be identified of material flow around which the flow of work-material results in a fold (refer to Fig. 4.7). The simulation of flow is conducted till the initiation of the fold; the process is then arrested with a view to re-meshing the deformed mesh and to adding sliding elements at the location of the initiation of the fold. One of neighboring surfaces around the dead point may be defined as "master surface" with reference to which the sliding elements may be arranged (e.g., surface A in Fig. 4.7). Another contact surface (B in Fig. 4.7) may be defined as "slave surface" with reference to which the sliding boundary is defined using the nodes on this surface. The definition of the slave surface should include sufficient nodes for the description of an area over which the master surface may fold. The simulation is then continued in a manner that allows contact analysis between free surfaces.

In comparison with the simulation of contact between a rigid body and a deforming body, the treatment of a deforming body as a contact boundary (slave surface) in a large sliding analysis results in the instability, since the slave surface itself may be undergoing large plastic deformation. It is therefore necessary to use small iteration increments to control the contact analysis; iteration parameters have to be subjected to repeated tests to verify the approach.

The comparison of simulations with experimental results suggests that sufficient accuracy was achieved [12]. Such simulations of folding flaws, while accounting for sliding of the work-material on itself, provide a more realistic pattern of material flow during injection forging. Simulation, in which adjacent surfaces of a fold are treated as homogeneously continuous, would undoubtedly produce unrealistic patterns of material flow and strain distribution.

Quantification of Component-Form Errors Due to Die-Elasticity

In nett-form manufacturing, the geometry of the component is prescribed by the die-geometry. The deflection of the die during the forming operation invariably causes the departure of the component geometry from that of the die [57]; the geometric errors in the formed component can be determined by attending to the deformation of material during tool loading, tool unloading, and the removal of deformed material from die [58].

Referring to injection forging of a fly-wheel (Fig. 4.8), the nominal dimensions of the component are prescribed by the die-cavity geometry. The dimension of the component is, however, influenced by the deflection of the die and the injection chamber. At the final stage of the forming, when the material issuing into the die-cavity has been in contact with the lateral surfaces of the die, higher forming-pressures are usually applied to enable die-filling of corners. Under these pressures, the die expands radially and axially. The expansion results in a component of a larger dimension than that prescribed. Upon retraction of the punch, the work-material is compressed radially and axiallly as the die contracts elastically to the extent that the component may be plastically deformed. This deformed component would undergo

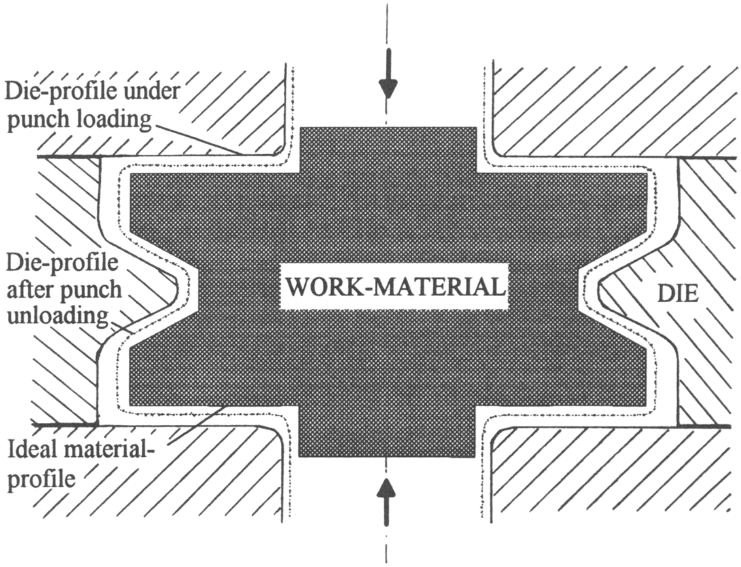

Die-profile under punch loading
Die-profile after punch unloading
WORK-MATERIAL
DIE
Ideal material-profile

FIGURE 4.8 Illustration of die-deflection and formulation of die-elasticity errors during the forming of a flywheel.

further elastic deflection when the component is ejected from the die. Computations have shown that different amounts of "elastic recovery" occur in different sections of the component.

Die-elasticity influences the form of the component; however, the extent of this influence is determined by a variety of factors [57, 58], some of which are the loading history on the forming machine, the elastic characteristics of the die- and work-material, the process parameter (temperature and friction), die geometry, and the construction of the tooling. These factors affect the component-form either individually or in combination; it is therefore important that the quantitative definition of component errors caused by die-elasticity is viewed in the correct perspective and in relation to the continually deforming work-material, which includes both the loading and unloading of tools and the ejection of the component.

Several modeling procedures may be used to analyze die-elasticity errors [57–60]. These are:

1. Elastic-plastic modeling regards both the die and the workpiece as elastic and elastic/plastic bodies respectively. Elastic-plastic modeling would enable prediction of pressure distribution on the die and enable a more efficient understanding of the variation of form of work-material during the forming cycle. This modeling would enable consideration of both punch loading and unloading, and treatment of the forming processes as non-steady-state-event. After each loading and unloading simulation, the extent of deformation of the material and die deflection are extracted to quantify dimensional variations; these variations can be understood further by animating the deformation/deflection and analyzing pressure-distribution on the die-surface.
2. Rigid-elastic/plastic modeling regards the die as perfectly rigid while the workpiece is subjected to an elastic-plastic analysis over the forming cycle. The analysis enables the computation of the pressure distribution on the die during material flow and, subsequently, the computed pressures are used to compute the die-deflection and stress distribution on the die. This modeling procedure would not enable consideration of the mutual influence of die- and work-material-elasticity on the form of work-material during tool-unloading.
3. Linear elastic modeling treats both the die and the workpiece as linear bodies and subjects these to the derived pressure-contours from experimental data. The model is simpler to use but requires the support of experimental data. The approach neglects the mutual influence of die- and work-material-elasticity on the form of work-material during tool-unloading, particularly when the work-material undergoes plastic deformation during unloading. This would result in large computational errors.

Other Modeling Considerations

Some particular aspects of modeling have to be taken into consideration prior to the application of FE simulation to the analysis of material flow. Generally, it is difficult to simulate material-flow for small aspect ratios of primary deformation zone (pdz) and fine exit-geometries because of spatial considerations for material deformation. One of the primary problems in simulation is to prevent severe distortion of the mesh, as this would result in inaccuracy of the solution and non-convergence of the simulation. If automatical re-meshing is not built into the software, special attention has to be paid to re-meshing in view of mesh-distortion which results from the rotation of the issuing material, surface folding, and the die-filling of sharp corners in the die-cavity.

In injection forging, the issuing material is deformed with a large gradient of strain at the exit from the injection-chamber. The situation is more severe the smaller the exit-radius. Simulation of flow past the exit-geometry requires the use of a fine mesh, which has been shown to be necessary for reinitiating the computation after re-meshing. Too coarse a mesh would cause severe discontinuity of the deformed mesh at the exit from the injection chamber and may prevent restarting the analysis. Particular considerations for initial meshing of the billet are also required to reduce the distortion of mesh; a dense mesh may be used in several strands of the middle elements in view of the large radial tension in the material in the core of billet (Fig. 4.9).

The simulation of unstable deformation requires the use of fine elements, as this would enable the smooth transition of the material from stable to unstable deformation. If coarse elements were used, simulation would fail when buckling was initiated. Incorrect buckling thresholds would also be predicted

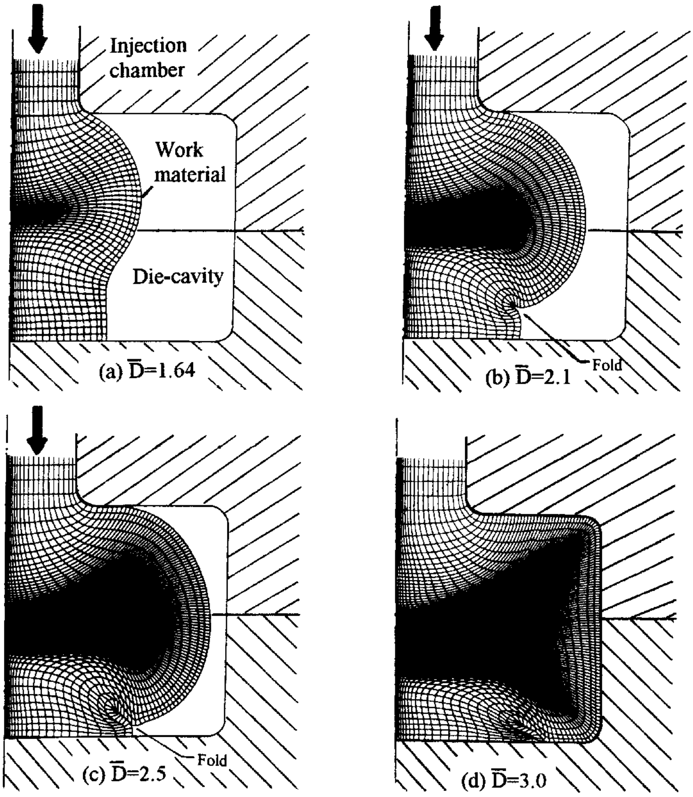


FIGURE 4.9 Simulation of injection forging of an annular flange.

due to inaccurate modeling. It is therefore necessary to test different meshing-schemes with reference to experimental results before selection of a suitable scheme. The stress analysis of a structure usually requires the use of high order elements to achieve the required accuracy of analysis; this is particularly true for the analysis of stress concentrations at particular locations. However, due to the fact that the mesh distortion invariably occurs in the simulation of large-scale material-flow in injection forging, lower order elements are recommended, as these are less sensitive to mesh distortion. The computational precision can be improved by using a fine mesh. Fine tolerances and large numbers of iterations should also be used for the simulation of unstable deformation. Numerical "uncertainty" is inevitable in the simulation of buckling and material folding, which usually occurs in the form of hunting between two contact point-pairs. Increasing the number of the iterations would also increase the chances of arresting hunting, which would also be affected by increasing the number of contact elements at key sections of the model.

Billet buckling, mesh distortion, and excessive use of contact elements may not be the only factors to cause the numerical uncertainty that may result in non-convergence of the solution; friction characteristics that are incorporated into contact analysis also have a significant influence on the convergence of the solution. Whether or not a friction model causes non-converging of contact analysis usually depends on the algorithm or formulation of the friction model that is to be incorporated in the FE software; this, incidentally, is an additional criterion for evaluation of the robustness of FE software. In ABAQUS, Amonton's Law is used for the contact analysis; the use of a constant coefficient of friction did not result in any numerical difficulty in the simulation of the injection forging. In order to use user-defined friction models, a user subroutine was designed to link with ABAQUS. The simulations showed that contact analyses were sufficiently stable to converge in the prescribed iteration steps, which may suggest that the friction-formulation used in ABAQUS for contact analyses is sufficiently robust for the cases that were examined.

Considerations for the Selection of FE Software

The following aspects should be taken into consideration when choosing FE software for analyzing injection forging:

1. FE modeling is required to refer to sliding of the work-material in the injection chamber as well as flow in the die-cavity. It is therefore important to select an FE code which has the capability of dealing with large sliding-contact boundary problems.
2. Both rigid/plastic and elastic/plastic formulations should be qualified for the analysis. In view of the requirement of the analysis of component-form errors due to the die- and work-material-elasticity, an elastic/plastic (large deformation) formulation would be required.
3. The software should have functions that are able to simulate plastic post-buckling behavior of the work-material.
4. Friction modeling should be flexible enough to enable incorporation of user-defined friction models into FE analysis; contact analysis with consideration of friction should be of sufficient numerical robustness.
5. The software should retain capability of simulating the development of folds in work-materials.
6. Functions such as flexible pre- and post-processing and automatic remeshing are not priorities, but may bear some weight in the selection procedure.

FE Modeling Procedures

The procedure for FE modeling of a forming operation is dependent upon the configuration of the software; the following procedures are described with reference to the use of ABAQUS.

Injection Forging of Solid Billet

ABAQUS contains the mixture of element- and solution-types to meet the modeling requirements for the simulation of injection forging. Both the injection-chamber and the anvil (or die) may be modeled as rigid-bodies for material flow simulation (Figs. 4.6 and 4.9). For the calculation of die-stresses, these may be modeled as elastic-bodies. The punch may also be modeled as rigid body, while punch displacement is simulated by prescribing a rigid displacement of the punch. Figure 4.9 shows an FE model of injection forging of axisymmetrical billet that was generated for use with ABAQUS—the modeling was obtained by using Element CAX4R [56] for the billet and IRS21A contact elements for billet/injection-chamber and billet/die interfaces. The FE mesh in the injection chamber is only partially displayed due to space limitations. The model may also be modeled as plane-strain (Fig. 4.6) or 3D model (Fig. 4.10) for analyzing the stability of the billet.

Simulation of multi-stage injection forging, as in processing which involves preforming, requires the modification of the model at intermediate stages of the simulation; typical requirements are re-meshing and modification of boundary conditions of the model. For the forming process shown in Fig. 4.11, the procedure for simulating multi-stage forming would be initiated by extruding the material into a die-cavity that is described by a die-insert (Fig. 4.11a). Subsequently, the material is deformed to produce an initial geometry (preform) (Fig. 4.11b). The rigid-constraint, which represents the die-insert, is then removed from the model (Fig. 4.11c) to enable simulation of the second-stage of forming, which would require remeshing the preform (Fig. 4.11c) and then mapping residual stresses and strains sustained at the end of the preforming stage to continue the simulation. In the second stage of simulation, new boundary conditions would have to be added to the model to simulate material-flow (Fig. 4.11d).

Modeling should also be able to deal with different tool-kinematics, for example, the anvil may be moved simultaneously with the injection of material, while the injection chamber and the die may be displaced simultaneously; these configurations may be used to modify the applied-force characteristics

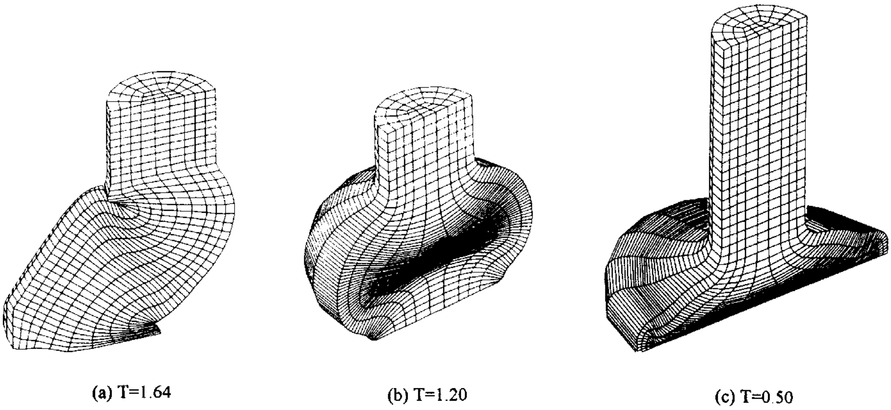
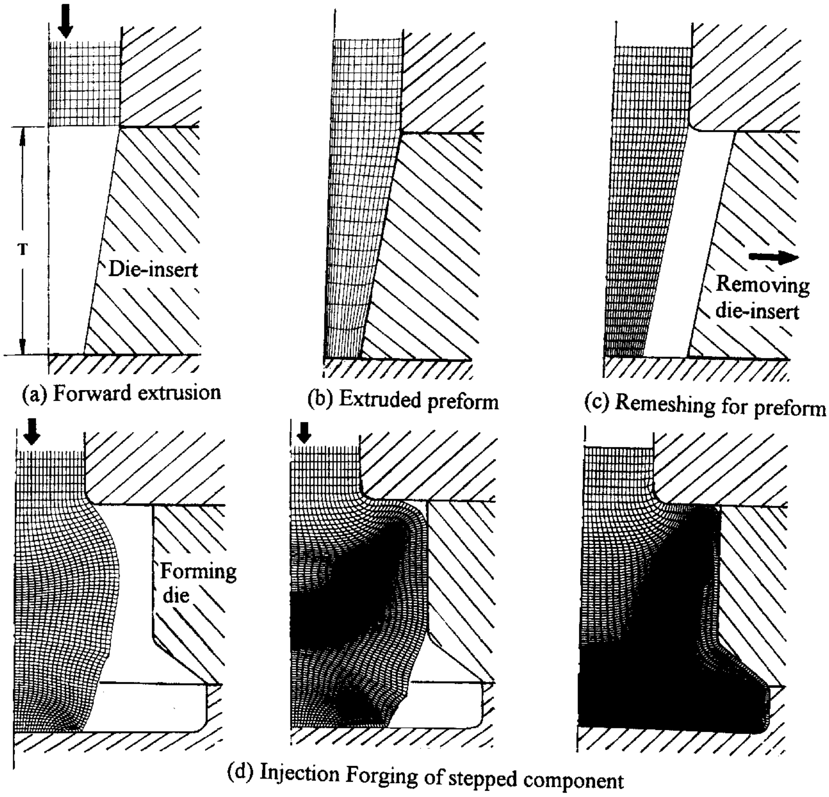

(a) T=1.64 (b) T=1.20 (c) T=0.50

FIGURE 4.10 Examples of 3D simulation of injection upsetting.



(a) Forward extrusion (b) Extruded preform (c) Remeshing for preform

(d) Injection Forging of stepped component

FIGURE 4.11 Simulation of the forming of a stepped component using a preforming procedure.

or material deformation mode. Simulation of the movement of tools can be effected by prescribing relative velocities for tools. For instance, in the simulation of injection forging with a retractable anvil (Fig. 4.12), the billet was initially injection-upset while the anvil was retained static (static boundary condition was applied to the anvil) (Figs. 4.12a and b). Initially, a free-gap height (parameter *T*) was used to prevent the development of folds, after which a prescribed relative-velocity was effected between the anvil and the injection chamber while the material was simultaneously injected into the die-cavity

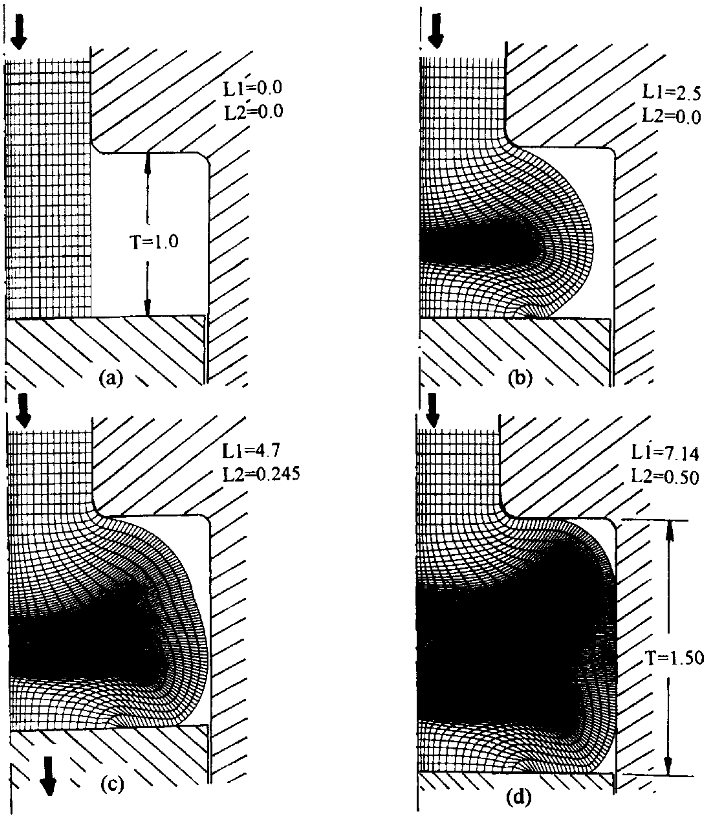


FIGURE 4.12 Simulation of injection forging with a retractable anvil.

(Fig. 4.12c). The process was continued after arresting the anvil at a position that corresponds to the required flange-thickness (Fig. 4.12d). The simulation of a complete forming-process requires several steps of numerical computation. Each step may require the modification of boundary conditions of the model. In such circumstance, however, this modification has to be attended without affecting the numerical stability of the model.

Injection Forging of Tubular Materials

Modeling of injection forging of tubular material with a supporting mandrel (refer to Fig. 4.2b) is similar to that for injection forging of solid billet. The mandrels may also be modeled as rigid bodies while the workpiece/mandrel interfaces are modeled using contact elements. Figure 4.13 shows an FE model and the simulated deformation mode for tubular material.

In the pressure-assisted injection forging configuration, the pressurisng-material (Fig. 4.14) is under pressures 10 to 20 times higher than their yield strength; these materials may be considered under hydrostatic stress conditions throughout the forming process. Though two different types of deforming materials are involved in the configuration, FE modeling of the deformation of the work-material may be simplified by treating the deformation as occurring while under uniform internal pressure. The pressurization may be modeled by prescribing values of hydrostatic pressure on the surface of the elements that are on the inside diameter and the injection of material by the displacement of nodes on the top-end of the tube. The deformation of the work-material may be simulated for several loading steps to develop the forming sequence.

In a manner similar to deformation during bulging of the thin-walled tubes (internal pressurization and free boundary-condition), the deformation of material during the injection forging of thick-walled

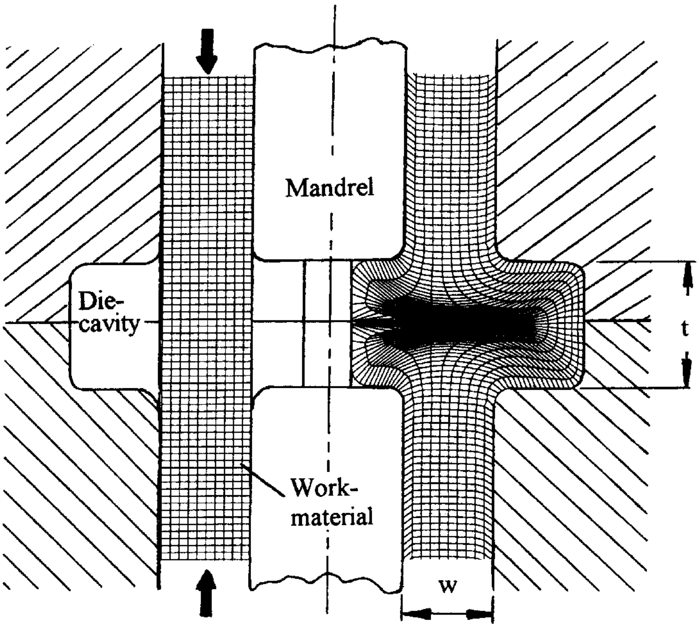
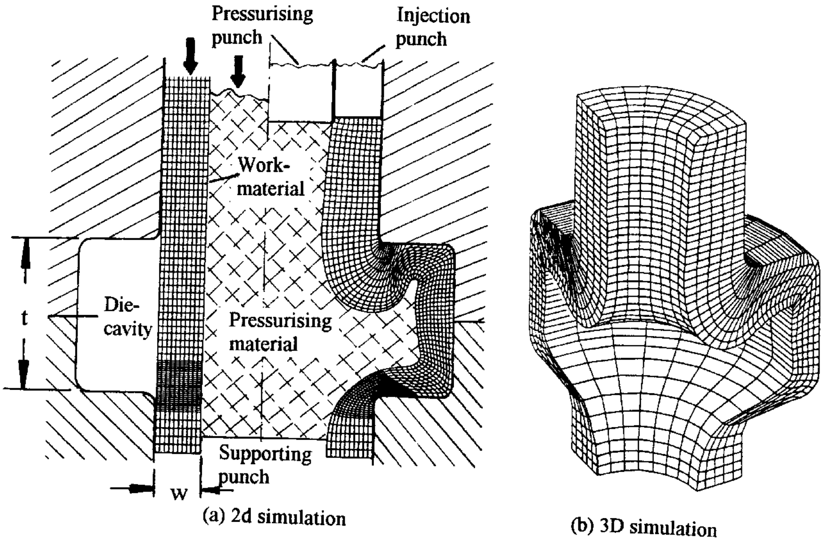


FIGURE 4.13 Simulation of injection forging of tubular material with mandrel support.

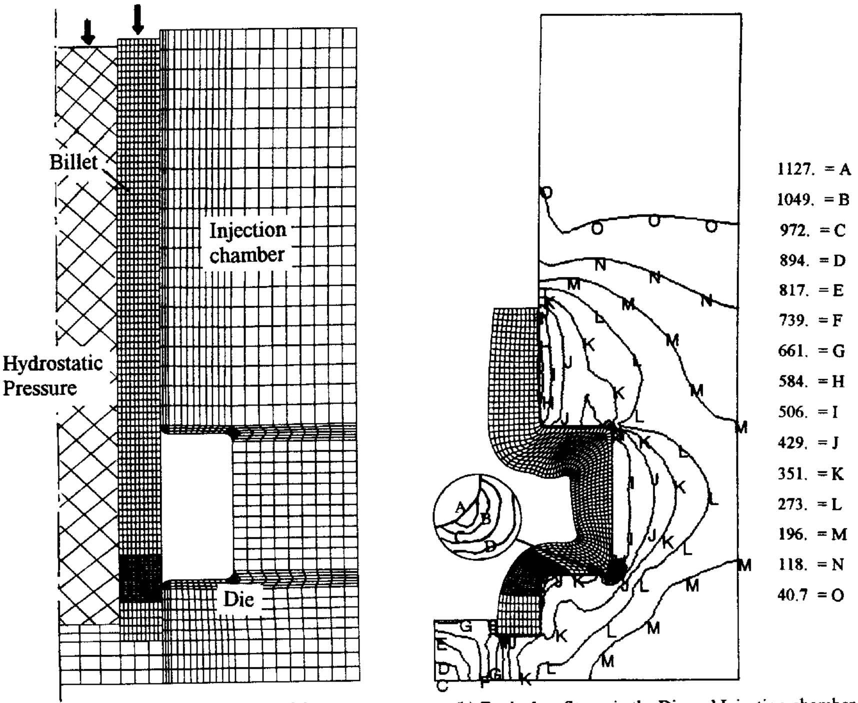


(a) 2d simulation

(b) 3D simulation

FIGURE 4.14 Simulation of pressure-assisted injection forging of tubular material with support of pressurizing medium.

tubes may be unstable for large pressurization and large aspect ratios (Fig. 4.14(a)). Due to the constraints applied by the injection-chamber and the die-cavity, the maximum load and the deformation mode is different from that for free deformation of tubes. Local necking of the tube at the exit from the injection chamber may initiate the buckling of the tube; this may occur before the material makes contact with the lateral surfaces of the die-cavity. The simulation of unstable deformation can be effected by overcoming the "numerical instability" that occurs in FE simulation.



(a) Meshes of the FE Model

(b) Equivalent Stress in the Die and Injection-chamber

FIGURE 4.15 Die-stress analysis of pressure-assisted injection forging of thick-walled tube.

The material deformation in pressure-assisted injection forging of thick-walled tubes is largely dependent upon the forming sequence—the combination of the pressure on the pressurizing medium and the injection of the work-material. In order to compare FE-simulated deformation with experimental results, FE simulation should be initiated with reference to the recorded force and displacement data of the two punches—pressurizing and injection. The verified models can then be used for developing complete simulations. For the simulation of the forming sequence, the dies may be modeled as rigid surfaces and the sliding of material along die-surface as rigid surface contact elements; this would enable a high computational efficiency. Simulation, in which the injection chamber and the die are treated as elastic bodies for stress analysis (Fig. 4.15), requires a greater amount of computing time.

4.4 The Applications of FE Modeling in Injection Forging—Examples

Analysis of Billet Stability

FE simulation was conducted to analyze the stability of the billet in injection forging [11, 52] using both 3D and plane strain models. The buckling of the billet is a 3D phenomenon; 3D plastic analysis , however, is time-consuming. The plane strain model was, therefore, used to enable an approximate analysis of the development of flaws. Benchmark tests were performed to enable the degree to which 2D models replicated buckling, prior to the extensive use of 2D simulation. The deformation and post-buckling behavior of billets were sufficiently similar in both 2D and 3D simulation and compared well with the failure forms observed in the experimental program to justify extensive use of 2D FE simulation.

Some examples of simulation results are shown in Figs. 6, 10, 16, and 17. Figures 4.16(a-1) to (a-3) show the deformation of the billet for aspect ratios of pdz of T = 2.5, 1.85, and 1.64 respectively. The

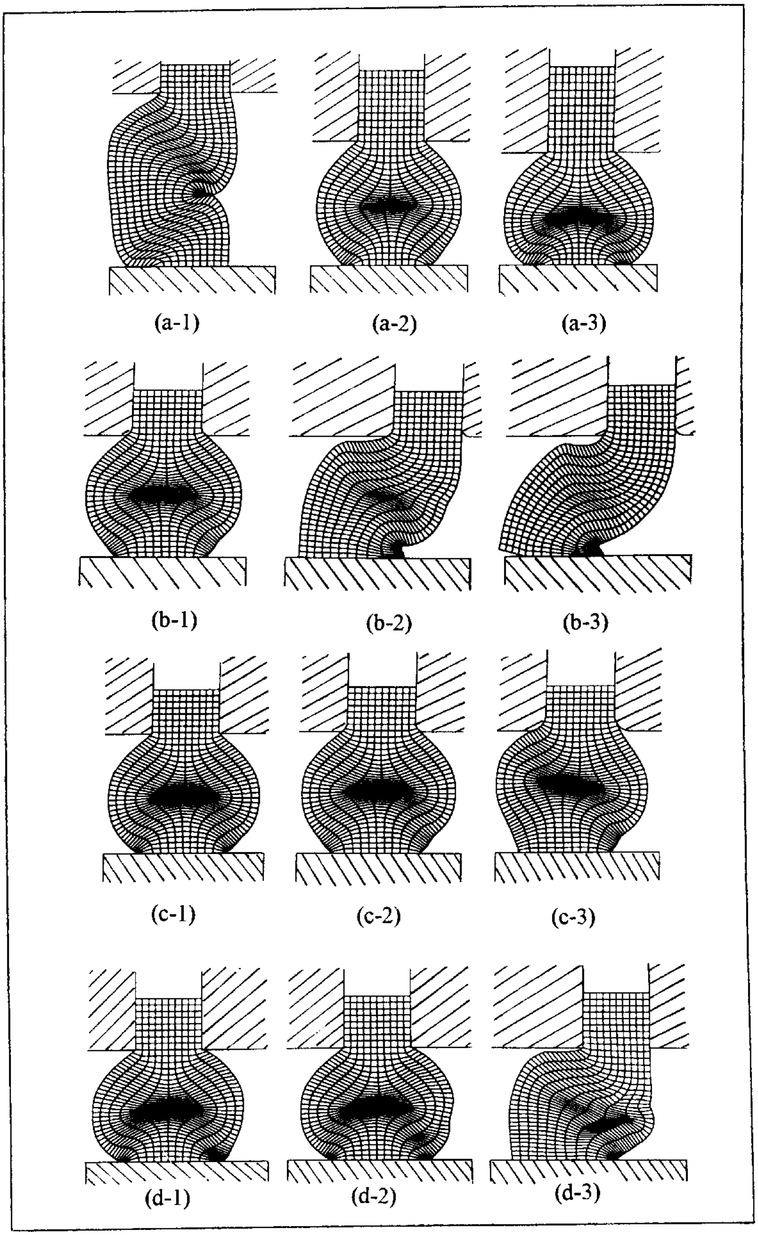


FIGURE 4.16 Simulation of the development of flaws due to the instability of billet.

billet, which was injected with an aspect ratio of 2.5, would invariably deform asymmetrically. Asymmetry was observed for T =1.85; this initial asymmetry does not make the billet unstable. The asymmetry would, however, result in the forming of a component which would not display symmetry in the flow patterns. By comparison with the simulation for T =1.85, that for T = 1.64 shows sufficient symmetry.

Simulation results for T =1.64 and coefficient of friction of 0.03, 0.02, and 0.01 are shown in Fig. 4.16(b-1) to (b-3). At μ = 0.03, the billet deforms symmetrically and continues to achieve die-filling. The stability of the billet is seriously affected when the coefficient of friction is reduced by a marginal amount. Under a friction condition of μ = 0.02 the base of the billet would slide laterally while retaining contact over its entire diameter. This would promote the development of a fold at the lower extremities

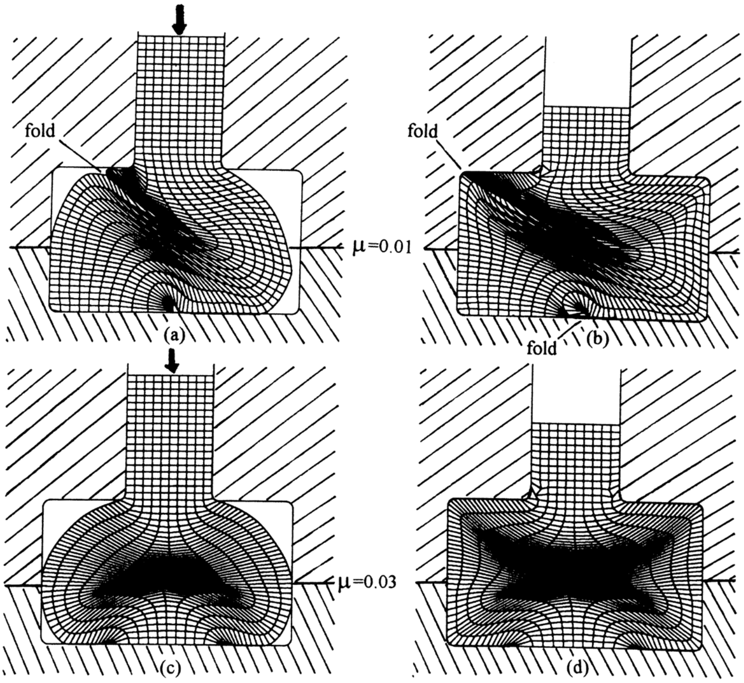


FIGURE 4.17 Examples of the influence of the friction on the forming of annular flanges.

of the billet as is shown in Fig. 4.16(b-2). When the value of friction coefficient is 0.01, the simulation shows that the lateral sliding of the base of the billet would be severe—refer to Fig. 4.16(b-3). This would cause part of the base of the billet to lift off the anvil.

The influence of friction on the development of flaws was simulated by injecting the material into a die-cavity of dimensions T = 1.64 and D = 3.0. The simulations for intermediate and final stages of the die-filling for friction conditions $\mu = 0.01$ and 0.03 are shown in Fig. 4.17. The billet was shown to be unstable ($\mu = 0.01$). The simulation of the deformation was continued until die-filling was achieved to enable the identification of flaws that develop under these friction condition. Figures 4.17a and b show the sequence of die-filling subsequent to the lateral sliding of the billet as depicted in Fig. 4.16; symmetrical die-filling was achieved by increasing interfacial friction (Figs. 4.17c and d). These specimens show that the profiles of the deformed material are sufficiently symmetrical. The simulation and experimental results suggest that the billets are generally stable when the coefficient of friction at the billet/anvil interface is of order of 0.03 for T < 1.65.

The influence of the exit-geometry was evaluated by examining three radii: $R = r/d_0 = 0.00$, 0.126, and 0.24 (Figs. 4.16(c-1) to (c-3)). The exit-geometry cannot be considered in isolation as increases in exit-geometry result in an effective aspect ratio of the pdz, which is higher than that derived from flange thickness. Using a larger exit-radii would improve the flow of material at the exit and reduce punch pressure; the incorporation of exit-radii, however, makes the billet less stable and hence more likely to deform in a manner that will result in the initiation of flaws (Fig. 4.16(c-3)).

Occasionally, batches of work-material are not metallurgically homogenous; the material property was not uniform across the billet, this being concluded since the grain size was not uniform. Examination of macro-etched ends of the commercially pure Aluminum billets revealed that a portion of it was of a fibrous-form while the remainder was coarse-grained. Simulation indicates that variation in metallurgical symmetry will result in asymmetric deformation—refer to Figs. 4.16(d-1) and (d-2)—and in extreme cases of inhomogeneity—refer to Fig. 4.16(d-3)—will result in the sliding of the billet.

Definition of Limiting pdz Aspect Ratios

Symmetrical deformation could be sustained for injection forging of a solid billet without end-constraint but with a high interface friction, at low aspect ratios ($T < 1.65$). Nevertheless, the experiments [10] showed that at $T < 1.65$, the material underwent a complex deformation mechanism; an intermittent type of shear, referred to as θ-shear, occurred during die-filling. Experimental evidence of this event indicates that even when the billet is stable, the quality of the component is also affected by flow-related phenomenon.

The simulation of material-flow in injection forging of solid billets was conducted with reference to three major process-form: injection forging in which the base of the billet was not radially constrained (Fig. 4.9); injection forging in which the base of the billet was radially constrained (Fig. 4.18); and double-ended injection forging in which the material was injected into die-cavity simultaneously in two opposite directions (Fig. 4.19). The typical patterns of material flow in these process configurations were extracted from experimental and FE results to enable the characterization of material-flow.

For injection forging without end-constraint, FE simulation results showed that folding of the free-surface of the issuing material was initiated close to the end of the billet (Figs. 4.7 and 4.9); this has been shown to occur when the aspect ratios of the primary deformation zone were greater than 1.2. For a die-cavity in which the base of billet was only partially supported, this threshold could be smaller [12].

Similar material-folds were also found in injection forging with an end-constraint and double-ended injection forging for aspect ratios of the pdz for which the flow of material was shown to be stable (Figs. 4. 18 and 4.19). During injection forging of solid billets with large pdz aspect ratios, folding defects develop

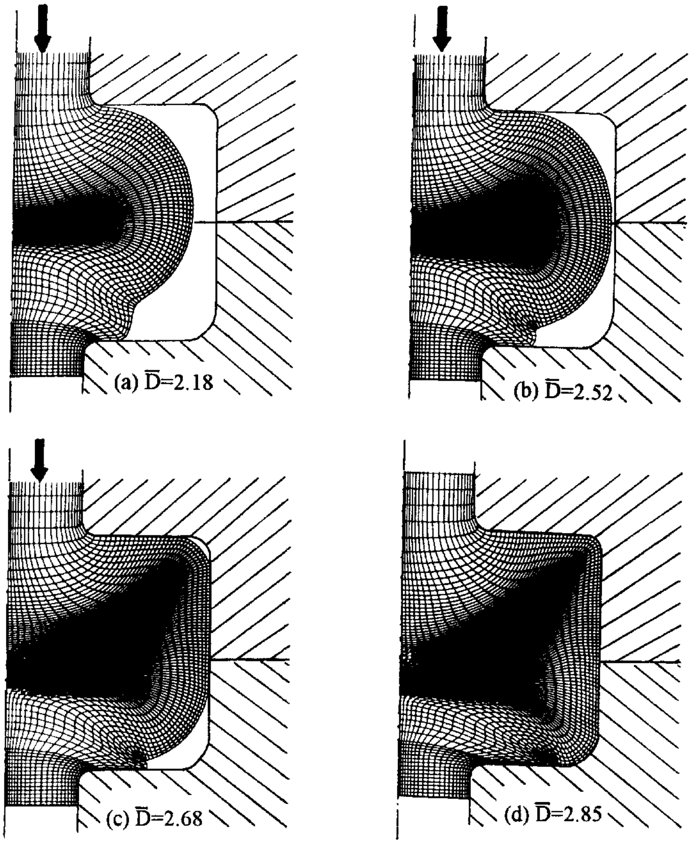


FIGURE 4.18 Simulation of injection forging of annular flanges with an end-constraint.

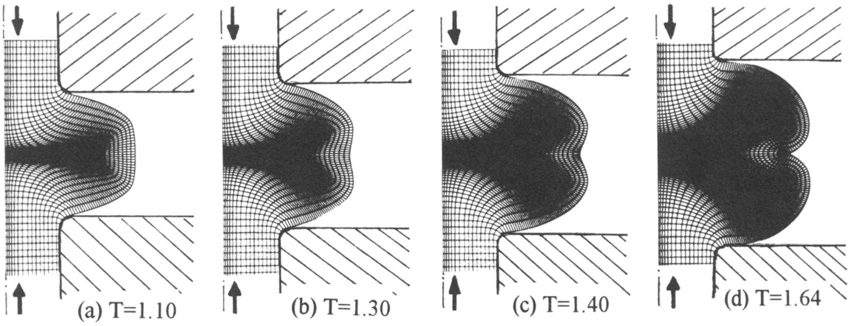


FIGURE 4.19 Simulation of double-ended injection forging for different *T* values.

as a result of the simultaneous rotation of the material issuing from the injection chamber and the radial expansion of the base of the billet. There is a prospect that if the base of the billet was not allowed to expand, folding might be eliminated. Injection forging with end-constraint is used to produce profiles in intermediate sections of the billet or to improve the stability of the billet during injection. With this configuration, the expansion of the lower section of billet is prevented. FE simulation results (Fig. 4.18), however, show that for larger aspect ratios, the constraint to the radial expansion could function efficiently only at the early stages of the injection. Continuation of injection promoted the flow of material onto the anvil-surface; subsequently, the material on the anvil expanded further. This expansion, together with the rotation of material, resulted in folds. In fact, the constraint to the billet only delays the initiation of the fold and would not eliminate it. By comparison with injection forging without end-constraint, injection forging with end-constraint only showed marginal improvement in the practicable range of the process.

FE simulation replicated the experimental evidence of folding, which was previously reported for double-ended injection forging of solid billet [62]. FE simulations show that these folds could also develop for $T = 1.3$ (Fig. 4.19). The end profile of the flange is only convex for aspect ratios of $T < 1.1$ to 1.2. For the profiles which are not convex, a fold or notch would eventually develop. In comparison with single-ended injection forging, the injection of material simultaneously in two opposite directions increases the possibility of folding. Similar conclusions may also be drawn for the injection forging of tubular material with mandrel support (Fig. 4.13). A major limitation of the application of this configuration (Fig. 4.13) is the folding of the material for large pdz aspect ratios ($t/w > 1.6$ to 18). The configuration which allows the inward flow of material (Fig. 4.13) may delay the incidence of folds; the development of the fold is a function of the t/w ratio, inner die-cavity geometry, and the extent of injection [20, 21]. Although the forming limit for this configuration is generally defined as $t/w = 1.6$, a fold may still develop after the inner-cavity has been filled; this would occur even for $t/w = 1.0$ for a larger extent of the injection.

The findings for both process-configurations indicate that the practicable process-range for injection forging should be defined with reference not only to the stability of the billet but also to flow which is axisymmetrical (billet is stable). Currently available results [12] demonstrate that aspect ratios for flawless injection forging are somewhat less than that defined by previous research. A threshold, beyond which folding would occur in injection forging, may be defined as $T = 1.2$ to 1.3.

Preform Design for Injection Forging

The development of flow dependent flaws during injection forging of solid billets indicates that the process-range, with reference to the stability of the billet, is $T = 1.64$. This range would be reduced to $T = 1.2$ due to the incidence of folds which occur in the range of $1.2 < T < 1.64$. The development of flaws reduces the process-range for injection forging and limits the forming of components to small aspect ratios of the pdz; this may be extended by introducing a preforming stage.

Several FE simulation techniques have been used to design preforms with a view to excluding material-flow defects in closed-die forgings [35]. Simulation of flow of material for a selected sequence of preform geometries may be simulated with reference to specific material flaws to derive the "optimal" preform geometries. The approach, also called "forward simulation," requires several iterations with different probable preform-geometires. An automatic backward tracing scheme using FEM was proposed [35], in which reverse simulation of die-filling was conducted by tracing the loading path of the forming process backward from a prescribed final configuration. Application of the approach included the preform design for shell nosing operations, the design of edge profiles for the reduction of cropping losses in rolling, and the preform design for the forging of H-shaped components and airfoil sections. Since the approach relies on an assumed strain-distribution which formed the basis for the reverse loading path, the result depended largely on the accuracy of this assumption. Further, for the forming of a complex component, the "backward tracing" approach would result in too many intermediate stages with die-geometries which may not be feasible from a manufacturing viewpoint. Recent research [63] has resulted in significant improvements in the case of moving and stationary boundaries (tools). Further effort is required to refine the approach with a view to replacing conventional preform-design approaches.

Although preform design using FE simulation has been applied to several forming processes [35], these retain characteristics which differ from those of injection forging. The approaches and strategies used in these applications may not be applicable to injection forging. In injection forging, preform design has to be conducted with reference to three considerations: material-folding, billet stability, and discontinuity in material-deformation in multi-stage forming. These were not addressed in previous research on preform design using FEM.

FE simulation and experiments were conducted to identify approaches by which the initiation of folds could be prevented, and hence improve the range of the process of injection forging. The experimental investigation on the material flow for different aspect ratios of pdz enabled selection of two approaches for the prevention of folds. Injection forging with a retractable anvil was proposed with a view to gradually increase the aspect ratio of the pdz during the injection of material (Fig. 4.12) as this would reduce the extent of the rotation of material in the die-cavity. The alternative was to use a preform with a view to reducing the expansion rate of the base of the billet during injection forging (Fig. 4.11).

FE simulation of injection forging with a retractable anvil indicates that the proposed configuration was feasible for producing components with large pdz aspect ratios ($T > 1.3$) without initiating flaws. The simulation indicates that the initiation of injection with $T = 1.0$ enabled the cylindrical surface of the billet to roll on to the anvil without initiating folds (Fig. 4.12(b)); the base of the billet had not expanded sufficiently to promote folding. The rotation of the material was small at the second stage of injection due to the formation of a large base during the forming of the component (Fig. 4.12(c)). Subsequent injection of the material allowed die-filling without significant rotation of the issuing material; hence, a fold was not initiated (Fig. 4.12(d)). In addition, since the rotation of the work-material was insignificant, die-filling was more symmetrical with reference to corners of the die. The corners of the die-cavity would fill simultaneously. Asymmetrical die-filling required higher injection pressures to effect filling of corners of the die. The simulation also showed that injection forging with a retractable anvil would not always result in successful forming. An appropriate relative velocity between the punch and the anvil must be maintained for effective forming; there is an optimal rate of injection to prevent the initiation of folds.

The approach using preforming involves at least two stages of forming. The first of these would be effected with a die-insert (Fig. 4.11(a)), which enables the forming of a frustum on the forward end of the billet (Fig. 4.11(b)). The preforming tool was then dismantled (Fig. 4.11(c)) and replaced either by spacers for injection upsetting or by a closed-die for injection forging (Fig. 4.11(d)). The preformed billet was not removed from the injection-chamber after preforming and was subjected to the second-stage injection after re-assembly of tools. As an alternative, the frustum at the leading end of the billet could be machined prior to forging; the preshaped billet was subjected to either injection upsetting or injection forging to enable the comparison of the effectiveness of preventing folds. Both FE simulations and experiments show that the frustrum-shaped preform may be used to prevent folds; the range of the process for injection forging of solid billets can thus be extended to T = 1.4 to 1.5 by using a machined preform and to T = 1.50 to 1.64 by using a preformed billet.

Forming Sequence Design and Die-stress Analysis for Pressure-assisted Injection Forging of Thick-walled Tubes

FE simulation and experiments were conducted to investigate the development of material-flow defects in pressure-assisted injection forging of thick-walled tubes to quantify the influence of individual process-parameters on these flaws [5, 25] (Fig. 4.14). Based on these studies, the forming-sequence was optimized with a view to achieving components with "uniform" wall-thickness. The forming sequence consists of the simultaneous pressurization and injection of work material in three stages. Initially, a small pressure was applied to the to ensure that the tube was supported uniformly; this was essential to prevent the material from buckling at the initiation of injection. Simultaneous internal pressurization and injection of the work material was then initiated to prevent the issuing material from folding while effecting die-filling. Filling of the die corners and the ironing-out of the inside surfaces of the component was effected at the final stage of injection by a substantial increase of pressure on the pressurizing material.

The simulation results, some examples of which are shown in Fig. 4.20, suggest that the pressurization at the second stage had an optimal value—too low a value would allow the development of a fold due to the upsetting of the tube and complete die-filling may not be achieved. A high pressurization would lead to thinning of the tube; this resulted as the tensile component of stresses in the work-material was high while injection pressure was relatively small. Because of the matched radial expansion of the tube under high pressurization, the material made early contact with the surface "A" of the die-cavity (refer to Fig. 4.20a). It is difficult for material which is subsequently injected into die-cavity to flow along this surface; the radial pressure exerted on the work-material is insufficient to effect the flow of the material along this surface. As a result, the material accumulated at the exit of injection-chamber.

Two further factors which influence the quality of the component are the volume of material injected to compensate for changes of tube thickness and the sequence of pressurization and injection. Figure 4. 20b shows an example of material deformation due to insufficient injection. The thickness of the tube will reduce with radial expansion, thus reduction may be compensated for by an increased rate of injection. Should reduction of the wall of the tube occur, it may not be compensated for at a later stage of die-filling, since the material injected at this stage did not contribute to more effective die-filling.

Simulation results showed that the quality of the component in pressure-assisted injection forging is largely dependent on the forming sequence. The performance requirement of a tubular component may demand uniformity of wall-thickness; this, in turn, would influence the prescription of forming sequence. It would be difficult to form fine corner radii, especially for thick-walled components. Due to the difficulty of causing material to flow along die-surfaces at later stage of the operation, the filling of the die-corners has to be effected by drawing material which is already contained in the die-cavity. As a result, the material at the section x-x (Fig. 20(a-4)) was ironed out by increasing pressure on the pressurizing material, the consequence being that a weakened section was produced in this area.

The forming sequence was "optimized" by comparing different combinations of pressurization and injection—Fig. 4.20c shows the "optimized" sequence. The approach that was adopted for eliminating the notch at the exit from injection-chamber and achieving "uniform" wall-thickness required that a low value of pressurization and relative large volume of the injection of work-material were effected initially to retain the wall-thickness as close to its original thickness as possible. The thickened wall prevented a weak section from developing at the exit from injection-chamber during subsequent forming as the filling of the corners could be effected by deforming the accumulated material into the corners of the die-cavity. The reduction of the wall-thickness during the final stage was not significant.

The development of pressure-assisted injection forging was based on the premise that the applied force would be reduced and, consequently, that the injection parameters would be of a lower magnitude. To confirm this assumption, a solid billet was injected to fill the same die-cavity. Both, the experimental and FE simulation results [26, 27] confirmed that the applied stress and energy requirements for the

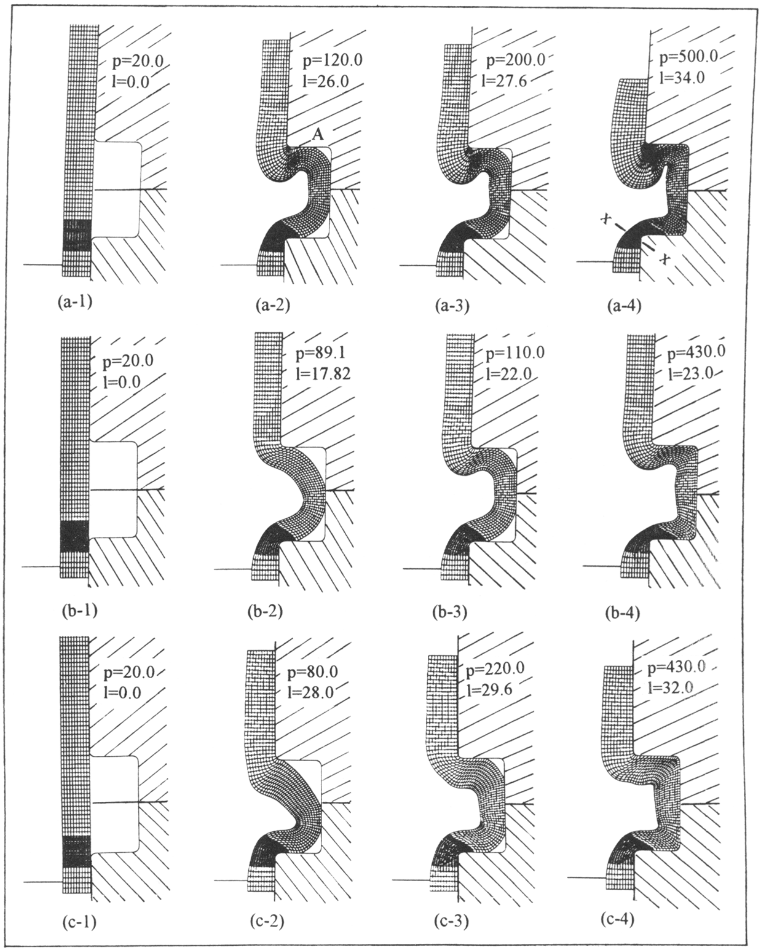


FIGURE 4.20 Illustration of the influence of forming sequence on material-deformation during pressure-assisted injection forging of thick-walled tube.

forming of solid billet for identical conditions—similar outer-diameter of the billet and aspect ratio of pdz ($T = t/d_0$)—were twice that for forming a tubular billet. The energy saving of this order may not be of consequence as it may not feature prominently in the overall economics of engineering processes. However, the reduction of applied stresses by the injection of tubular material would improve die-life. Replacement of metal-forming dies constitutes a major cost in the economic balance of processes.

Forming using the sequence shown in Fig. 4.20c suggests that the level of stresses in the die and injection-chamber was of the order of 700 MPa for aluminum specimens of the dimensions shown in Fig. 4.15. This stress is well within the working range of metal-forming dies. Taking into account the fact that the stress on the tools is dependent upon the forming sequence and that neither the maximum force nor the displacement of both punches provided an indication of the extent of die-filling at the final stage of the process, FE simulation would be an efficient approach for designing the process, with a view to preventing excessive tool stress.

A simulation of the injection forging of tubular, low carbon steel with wall-thickness of 5mm indicated that the injection pressure would be approximately 2000MPa, which imposes tool-stresses of 2500 to 3000 MPa in the dies for the forming-sequences examined. This is in excess of the working strength of die-materials used. A warm forging process configuration would reduce this to tolerable levels, for which a metallic pressurizing material would have to be used.

4.5 Computer-Aided Design and Manufacture of Injection Forging

The design activity is responsible not only for the performance and appearance of the product but also for the cost of the component. Design, therefore, cannot be an isolated activity but must address all available manufacturing routes, with a view to optimizing the quality and cost of the component. With reference to nett-forming, the design exercise is conducted not only to specify the component-form but also to address all manufacturing constraints—machine, material, tooling, and processing conditions.

Computer-aided "design for manufacture" is currently the main form of implementing of the "concurrent engineering." To enable this, CAD/CAM is popularly used as a design approach. Using CAD/CAM approaches, simultaneous design would be effected efficiently by supporting the designer with information on all possible resources required for the design and manufacture of components. Some CAD/CAM systems [42] have demonstrated the potential for the development into decision-support systems for component/tool design.

Computer-aided design and manufacture for nett-forming by injection forging is being developed as an aspect of research associated with the development of a decision-support system [64].

Methodology

In order to develop a decision-support system for component/tool design using a CAD/CAM approach, several design/evaluation methods have been developed [58, 60, 64–68]. These are described briefly in the following texts.

Geometric Modeling

The popular strategy used for the development of the design-support systems for forging was to evolve a 2D-CAD system for component and tool design. The system was linked to a knowledge-based system to enable the evaluation of manufacturability. Subsequent to the evaluation of the geometry, the component was transferred to a CAD software to enable detailed design. This approach required the design to operate in several software environments. An integrated system, supported by solid modeling, would enable design and assessment of a component more efficiently. A solid modeling-approach—principal feature modeling—was used to enable component-design for forging within a solid modeling environment [65, 66]; the approach enables integration of currently available 2D-based knowledge-based systems.

Design for manufacture requires that the component form is specified in a modular form in order to enable the evaluation of the design. The component may be defined as a combination of primitive forms as is the case in "design by features;" alternatively, the primitive forms which constitute the component may be extracted and identified automatically. Unfortunately, both these approaches are currently at a stage of refinement which only allows their applications to a limited range of component forms. Principal feature modeling [67] combines the strategies of both "design by feature" and "feature recognition" to enable efficient modeling and feature manipulation; the approach was proven to be particularly efficient for the modeling of forging/machining components [65]. Designing is attended to with reference to a prescribed set of performance requirements rather than to prescribed form features. The principal features, which represent the principal geometric profiles of a component, may be defined by the designer using arbitrary geometry—a group of curves on a plane or a curved surface. The principal features which have been generated are linked, exclusively, to a set of prescribed attributes which are catalogued in a database.

The solid model of the component may be created by geometric manipulation of a principal feature; several principal features may be defined for more complex components. Subsequent to the definition of the principal features for a particular component; local features may be added to produce the first iteration of the "performance" model of the component. The form of the additional features is generated by the modification of the principal geometric entities; these additional features are extracted and classified as individual manufacturing features. In circumstances where components cannot be modeled by defining principal features, the model can be created by other approaches. This will enable the prescription of the principal feature by the extraction of the curves on a feature plane, which is prescribed by the designer. In comparison with the "design by feature" approach, the proposed approach [67] does not require a pre-defined feature library that would constrain the flexibility of the system. Further, the difficulty and complexity of the routines for the recognition of form features are reduced since the principal features, which are recorded in the system, provide auxiliary geometric and topologic information of the component model; this simplifies the recognition of manufacturing features.

Classification of Components and Forming-Features

Component-forms which may be injection forged can be classified using the approach shown in Fig. 4.5. These forms are based on a classification of the injection forging processes—injection forging of solid billet, tubular material with mandrel support, and pressurizing medium (Fig. 4.2). The major component-forms are flanged and branched forms. These are essentially variations of geometry in the radial direction. Components with geometrical-variations in axial direction can also be produced simultaneously with the forming of radial geometric elements when injection forging is combined with forward- and back-extrusion. The three process configurations shown in Fig. 4.2 also indicate that flanges can be either solid or hollow, and components can be formed with solid and tubular bodies. Further, the flanges and branches may be formed inside, outside, or on both sides of tubular material; the axial geometric elements may be formed simultaneously in single or both directions. These variations of injection-forged component-forms may be classified into seventeen families (Fig. 4.5). Based on these families and considering all possible variations of the distribution and geometric-form of formed elements, the component-forms may be further divided into some sub-classes. These include the number, appearance, orientation, and position of these forms relative to the axis of the billet. The flanges may contain flat, concave, or convex features and the branches may also be solid, cup, or tube form; these may be distributed rotational-symmetrically, face-symmetrically, or asymmetrically. Based on this classification (Fig. 4.5), a database is currently being created to contain design-knowledge of process configurations and component-forms, to support the development of a design-support system for injection forging.

Based on the proposed classification of components, the forming-features of components, which are to be evaluated, are classified and are represented in databases using the architecture shown in Fig. 4.21. Each principal feature and additional feature is decomposed into individual geometric elements. Based on this decomposition, three types of feature parameters are derived by applying geometric reasoning to surfaces, edges, and points of these geometric elements. These are:

(a) Flow-dependent features such as corner radius, surface draft.
(b) Stability-dependent features such as aspect ratios and section ratios.
(c) Volume-dependent features such as sectional area and indicative signs.

Subsequent to component-design, which includes the generation of the geometric model, the assessment of the component, tool selection (or design), and process sequence design, the following parameters are attached to the component as attributes. These parameters relate to processes which will be used for the transformation of the material:

(a) Manufacturing specifications: tolerances, surface texture, component properties.
(b) Work material: materials codes (material library).

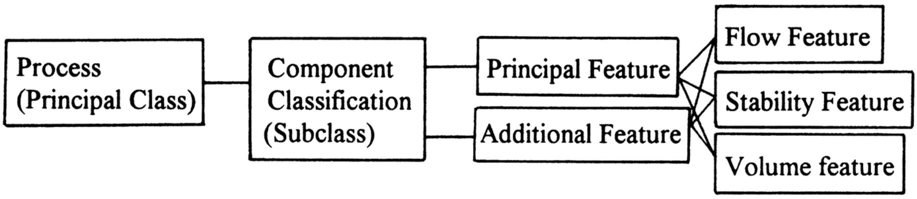


FIGURE 4.21 Structure of the representation of component and features for forming.

(c) Lubricant types: lubricant codes (lubricant library).
(d) Processing conditions: working temperature.
(e) Forming sequence: tool codes (tool library), preform codes (preform geometry library), and forming sequence-number.
(f) Forming Press: press codes (machinery library).
(g) Post-processing: heat and surface treatment (material-finishing library).

All information is linked to the model of the component and can be inquired and modified by manipulating the attributes of the model.

Component-Form Error Analysis

The geometry of components, which are manufactured by the bulk-deformation of materials, is process-dependent. The designer would have to account for processing constraints prior to determining the form and tolerances on the component. Current requirements for the nett-forming of components necessitate the definition of both processing sequence and tools design, with reference to form-errors.

Automatic form-error analysis of components was developed to support the design and assessment of components and tools for forging [59, 66]. Form comparison is based on comparisons of solid models of the component with that derived from FE analysis and that measured using a Co-ordinate Measuring Machine (CMM). These comparisons enable the generation of an integrated information model, which can be used to define tool-compensation requirements. During error-analysis, the specified surfaces or features are extracted individually from the specific model of the component with the view to comparing corresponding points and surfaces. These points may be either FE nodes with deformation values or measured points that are imported for automatic classification with respect to each manufactured surface; the errors are computed subsequently for each of the locations on the component. The computed errors and deviations with respect to specified tolerances are recorded in the database with reference to corresponding designed entities. The errors may then be displayed together with the specific model of the component and the specific surfaces by defining an error point, error line, or error surface display. The designer is then able to evaluate the designed model of the component by analyzing surfaces of interest.

Decision-Support for Component/Tool Design

A flexible evaluation routine is essential for design-support for component/tool design; the evaluation routine has to be both software-technology dependent (interface and graphic tools) and design-methodology dependent. Flexibility of design evaluation and broad-based decision-support are dominant objectives in structuring the decision-support-system for component/tool design. Three circumstances under which the designer needs support for decision-making during component design have been considered [64]. These are: (a) the designer's experience and knowledge is sufficient to interpret the requirements deal. Support, in the form of standard empirical data, would be sufficient; (b) the designer needs support in the form of knowledge, which refers to performance and processing constraints. In most cases this may refer to rules which are company-specific and expert-knowledge in the fields of design, assembly, process-planning, manufacturing, costing, and marketing; and (c) the component to be designed is new or has specifications for performance and processing which require experimental trials. In the developed system, the evaluation of the design is effected interactively by drawing from experimentally-proven data, knowledge, and process simulation.

Process simulation—The system development for process simulation for machining options has involved the integration of CAM software for tool manufacturing because the technologies and software in this technology domain have been relatively efficient. The design of the forged components must, however, contain the character of the deformation process. Such considerations do not usually feature in the design of a machined component. Further, subsequent to the definition of the component, it would be necessary to design the required tools at which stage the designer would be required to consider the structural characteristics of tools.

FE elastic analysis has been widely applied to the design of forging tools; to a lesser extent, plastic flow analysis has been used to support the design of components. Current shortcomings of FE analyses for plastic analysis refer to computational requirements. A knowledge-based system is, therefore, being developed for application to the injection forging, with a view to minimizing the dependence on simulation. Several FE simulation technologies have been developed for the design and assessment of forged components—some of these have been described in this text (refer to section **Particular Modeling Technologies**). Results from simulation which define flaws, forming limits, material requirements, component precision, and tool stresses are recorded in databases. These are linked to the specified models and referred to in the design of similar or new component-forms.

Expert system—In a decision-support system for component/tool design, available knowledge on performance and manufacturing constraints has to be represented by referring to each designed or manufactured form. This may be effected by referring to classified features for manufacturing. Each data and rule may be recorded with a link to standardized geometric form. Simultaneous reference to knowledge in multi-technology-domain was not easily achieved in earlier systems based on modeling or feature extraction. However, in the developed system, the knowledge is represented simultaneously for individual form features and for geometric elements for a component model, and also for the principal features of the model, which has advantages in assessing design concepts with reference to both global performance and local features.

The component/tool design system is "multi-technology-domain" which addresses forging and machining; machining is required for manufacturing tools to achieve the nett-form of engineering components. In its application to machining, the rules and standard data are represented in databases with reference to individual classified form features; the designed model is assessed by extracting and displaying individual form-features and their attributes, such as tolerance and finish. For complex component-forms with interacting features, the model may be assessed using manufacturing process simulation, which is conducted using an CAM package. With consideration for the particularity of component design for forging, knowledge is represented in the databases in object-oriented structure by following the architecture shown in the section **The classification of components and features for forming**. The databases are built on the classification of process, component-form, material, machine, tool, and lubricant. Relevant data and rules were gathered from several academic and industrial sources for the classification in a form amenable for computerization. The expert vocabulary has been modeled and structured, and expert knowledge is formalized with reference to both individual manufacturing features and principal features of the component. The evaluation of general components is conducted by means of rule-based reasoning which draws from both expert knowledge and the results of previous FE simulations; this is made possible by operating several operation modules for knowledge acquisition, reasoning, and knowledge renewal. The reasoning is also strengthened using Neural Networks for interpolation between proven forming sequences. The system only resorts to the use of the simulation of deformation when unfamiliar components, material, or forging conditions are encountered.

Neural networks—Assessment of performance and manufacturability of forging requires reference to material geometry, forming sequence and process conditions; these may not be familiar to the system since no previous knowledge exists. Further, FE simulation is time-consuming. Neural Networks are ideal for interpolation between proven component performance and forming sequence. A Multi Layer Perceptron (MLP) neural network with backpropagation learning algorithm [68] was proposed for preform design to achieve flawless forgings.

Neural networks may be used to map the geometric relationship between component and preform shape. Gray-scale image is used for the representation of the model of the component and preform. The model of the component and proven preform is represented by a black-white image and read directly by an image scanner. The data is standardized using 16 × 16 image-format through a series of image processing operations and the standardized component image is dispatched along with a shape complexity factor value as input to the neural network; the standardized preform shape data is the output. After rigorous training and testing, the system is used to support the design of preforms. The technology is also being developed for the prediction of component performance with reference to similar and proven component-forms.

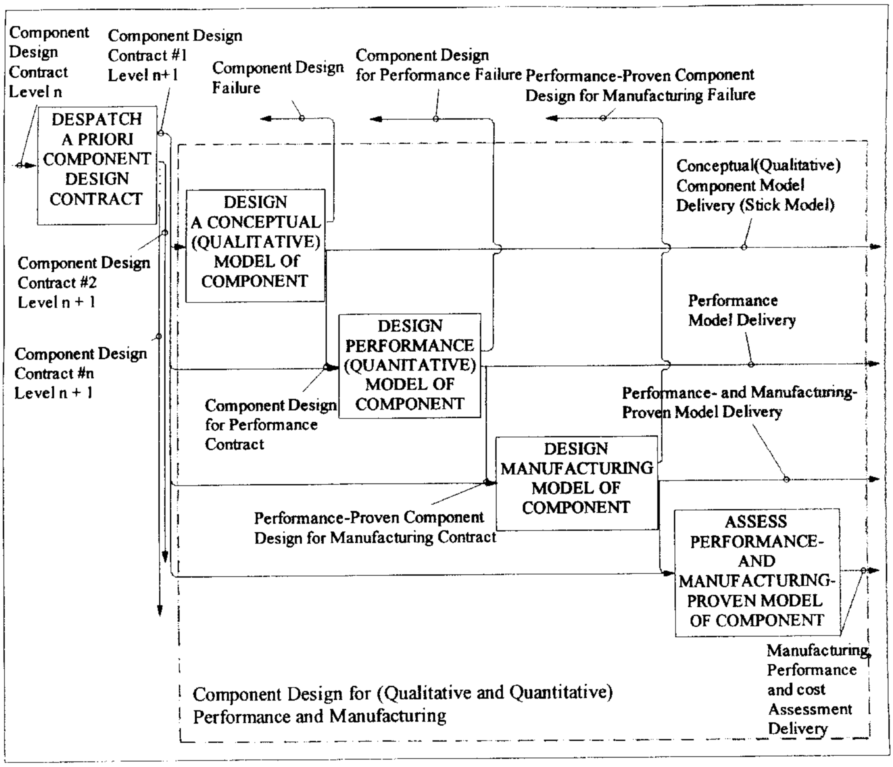

Component Design Contract Level n
Component Design Contract #1 Level n+1
Component Design Failure
Component Design for Performance Failure
Performance-Proven Component Design for Manufacturing Failure
DESPATCH A PRIORI COMPONENT DESIGN CONTRACT
DESIGN A CONCEPTUAL (QUALITATIVE) MODEL Of COMPONENT
Conceptual(Qualitative) Component Model Delivery (Stick Model)
Component Design Contract #2 Level n + 1
Component Design Contract #n Level n + 1
DESIGN PERFORMANCE (QUANITATIVE) MODEL OF COMPONENT
Performance Model Delivery
Component Design for Performance Contract
Performance- and Manufacturing-Proven Model Delivery
DESIGN MANUFACTURING MODEL OF COMPONENT
Performance-Proven Component Design for Manufacturing Contract
ASSESS PERFORMANCE-AND MANUFACTURING-PROVEN MODEL OF COMPONENT
Manufacturing, Performance and cost Assessment Delivery
Component Design for (Qualitative and Quantitative) Performance and Manufacturing

FIGURE 4.22 Structure of the design and assessment of component/tool.

Structure for Design and Assessment

The objective of the development of the system is to provide the designer with decision-support during component/tool design, which refers to the principal stages in the evaluation of the models of the component; these are conceptual, performance (environmental and operational), and manufacturing. During component design, the routine should not be monotonic, the design and evaluation of the model should be supported at all stages. With those considerations, the process of component/tool design may be structured as shown in Fig. 4.22.

The system accepts the input from the component design contract or an unfinished design; the design may then proceed from any stage. The model will be tested at every stage; the failed design will be returned to the previous stage to be modified or totally rejected due to its inability to match specified constraints. The satisfactory model is transferred to the next stage, which will be subjected to analysis in accordance with the requirement of this stage.

Concept Design—Computerization of the procedure for assessing alternative concepts with reference to the technical and commercial constraint is a formidable task. The qualitative evaluation, however, still can be conducted to the certain degree, for example, design documents of analogous products can be integrated to arrive at decisions. The assessment may also be initiated by process simulation to confirm the feasibility of a concept. At this stage, more than one concept may be generated. After sufficient comparison, the preferred solution is visualized using technical specifications and attributes.

Performance Design—This stage receives the component qualitative performance delivery from the concept design or a new component design for performance. Further, principal features are designed and local features are added. The designed model is tested according to the performance constraints; the constraints may be static or dynamic and qualitative or quantitative, which refer to features such as appearance, reliability, maintainability, disposability, etc. The difficulty of producing a universal module by which decisions may be made relative to constraints will require manual interaction as the means of designing with reference to performance constraints. The system is designed to enable easy access to the data of the designed model so that the designer can extract the necessary information (geometric or topological, point, line, face, or surface). Secondly, the system provides the standard subroutines by which the designer can develop the required formulation to judge constraints, such as dimensional limits of tools and positioning and interference analysis of tooling assembly. In addition, the module is also designed to provide interfaces to enable the use of analytical approaches.

Detailed Design—Before the detailed design of the manufacturing model is initiated, the features of the model are classified automatically and interactively. The classification is manufacturing method-based, i.e., for forging and machining, different classification strategies are used for defining manufacturing models. Geometric features, which have been classified, are displayed individually on the screen to enable definition of the manufacturing-related attributes.

The general requirements, such as the capacity of equipment, tool, and material, may be investigated in the concept design. The available CAM facility for tool interference and fixing verification may be used to test the finished tool-model; further detailed judgments of the manufacturability are conducted for some local and individual features. At this stage, the required FE simulation of material flow, analysis of tool deflection, and stress may be conducted. Further, the designer would also rely on machining simulation to compare the cost-implication of tool manufacturing, which results from the addition of features to the performance model. The precise shape, dimension, and tolerance of the designed tool-model are finally specified for the manufacturing models.

Assessment of Design—The accepted models, after detailed design, are a performance- and manufacturing-proven component design. In other words, the design satisfies the conditions of both performance and manufacturing constraints. Further assessment of the models is conducted using synthetic cost data; the cost of each model, the results of the assessment, are delivered as design documents, some of which may be recorded in a database of the system for reference in other designs, or used for adding and renewing the design knowledge in the library.

System Development Considerations

At the primary stage of the development of the decision-support system for component/tool design, the application concentrates mainly on the nett-forming by injection forging and machining for tool manufacturing. The cost of operations within each technology is qualified prior to the development of software to structure the decision-process, the system and each functional module.

Four major modules are planned; the simplified system-architecture is shown in Fig. 4.23. The CAD/CAM Module mainly includes basic CAD and CAM tools for component design; currently, ACIS [69] is chosen as geometric engine to provide primary CAD/CAM tools. ACIS is a new-generation, three-dimensional, boundary-representation, solid geometric modeler; it provides efficient functions and interfaces to enable flexible manipulation of geometric entities and their attributes. General CAD or CAD/CAM commercial packages are designed for direct application to design and manufacturing. ACIS, however, was designed mainly for fundamental research purposes; it is, therefore, more able to support users in implementing their own approaches and to develop application software. The Processing Module consists of the process-related data, knowledge, and process simulation sub-modules; the Company-Specific Information Module contains the company-specific information on design specifications, product, material, equipment, costing, etc. The Operational Module is an assembly of a number of sub-modules which includes feature extraction and attribute manipulation, pre-processing of the model, user-defined constraint analysis, component-form error analysis, model-assessment and synthesis, and the preparation of design documents. The modules may be individually and simultaneously operated through common interfaces and a central database system.

In the system, all data, both dynamic and static, will be retained in the several databases, each being designed for a different functional purpose. Since the decision-support for component/tool design will eventually be company-specific, the system is open for the designer to develop; the system provides flexible interfaces. The user will, therefore, be provided with a shell.

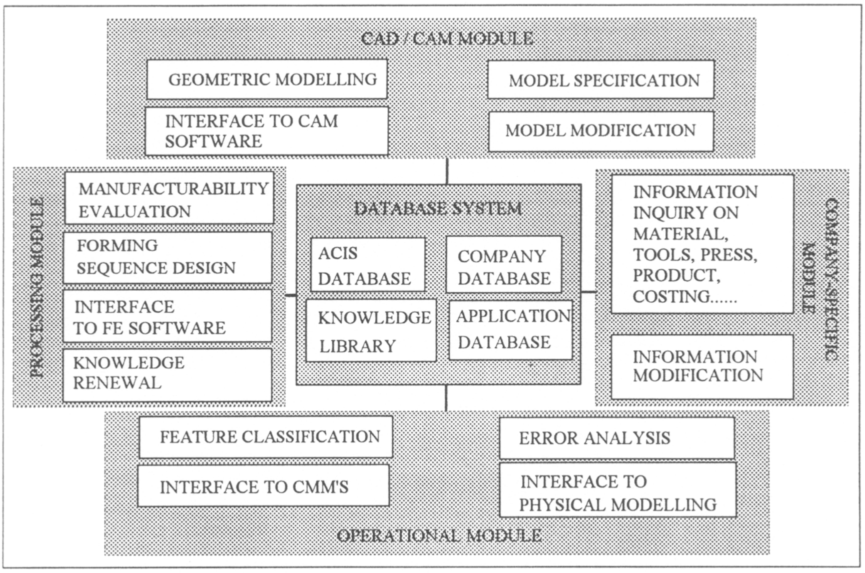


FIGURE 4.23 Software organization of decision-support system for component/tool design.

4.6 Conclusions

FEM has been applied to injection forging to analyze forming-force requirements, to simulate the development of flow-dependent flaws, to determine the limiting pdz aspect ratios, to design preforms to extend the process-range, to design the forming-sequences to improve component-quality, and to quantify component-form errors. These have been possible because several FE-modeling procedures were developed for injection forging. The success in analyzing injection forging suggests that FE is an efficient tool for the design of processes and tools to achieve nett-form of engineering components. Efficiency of the design and analysis will further be improved by using a decision-support system for component/tool design in which FE analysis, CAD, CAM, expert knowledge, and Neural Networks techniques are to be integrated. The integration of these functions will enable more efficient use of FE simulation for analyzing forming-processes and more efficient assessment of the design, so that the design will meet performance and manufacturing requirements.

References

1. Cogan, R. M., Hydrodynamic compression forging, *General Electric Company,* ASD Report 7-890 (111), 1963.
2. Geiger, R., Near-nett shape forming by radial extrusion, Proc. 7th Int. Cold Forging Congress, 24-26th, April, Birmingham, UK, p. 29, 1985.
3. Alexander, J. M. and Lengyel, B., "On the cold extrusion of flanges against high hydrostatic pressure", *J. of the Institute of Metals,* 93:137, 1965.
4. Aliev, I. S., Radial extrusion process, *Sov. Forg. Met. Stamping Technology,* Part 3, p. 54, 1988.
5. Balendra, R. and Qin, Y., Pressure-assisted Injection Forging of thick-walled tubes, *Int. J. of Mach. Tools Manfact.,* 35:11:1481, 1995.
6. Rozhkov, S. I. and Kirsanov, K.A., One- and two-directional extrusion of parts with one branched rod element, *Soviet Forging and Sheet Metal Forming Stamping Technology,* No. 3, p. 5, 1988.
7. Enghave, C., Tvaerflydepresning af roerflanger (in Danish), Procesteknisk Institut, DtH, MM-rapport ny. 91.08, 1991.
8. Osen, W., Variation, of the transverse impact extrusion process, *Wire*, 35:3:109, 1985.

9. Balendra, R., Pressure to effect injection forming, *Int. J. Adv. Manufact. Eng.,* 2:32, 1990.
10. Balendra, R., Injection-chamber to die-cavity interface for Injection forming, *Int. J. Mach. Tools Manufact.,* 33:6:753, 1993.
11. Balendra, R. and Qin, Y., FE simulation of the development of flaws during Injection Forging, *Int. J. of Mach. Tools Manfact.,* 34:8:1091, 1984.
12. Balendra, R. and Qin, Y., Material-flow considerations for the design of Injection Forging, *Transaction of ASME, Journal of Manufacturing Science and Engineering,* 119:350, 1997.
13. Kudo, H. and Shinozaki, K., Investigation into multiaxial extrusion process to form branched parts, *Proc. of the Int. Conf. Prod. Eng.,* Tokyo, Part I, p. 314, 1974.
14. Osen, W. and Lange, K., Deformation pattern, forces and process limits in combined radial and backward can extrusion, *Steel Research,* 56:4:211, 1985.
15. Osen, W., Possibilities and limitations of cold extrusion processes combined with radial extrusion, *Advanced Technology of Plasticity,* 1:575, 1986.
16. Krusic, V. and Kuzman, K., Cold radial extrusion in automotive industry, 26th ICFG Plenary Meeting, Osaka, 1993.
17. Arentoft, M., Petersen S. B., Rodriques, J. M. C., Martins, P. A. F., Balendra, R. and Wanheim, T., Review of the research on the Injection Forging of tubular materials, *J. Mater. Processing Technol.,* 52:2-4:460, 1995.
18. Henry, J. C. and Watkins, M. T., The production of hollow flanged components from bar stock by upsetting and extrusion, NEL Report 628, UK, 1977.
19. Okafor, A. C. and Balendra, R., Stress and energy distributions in injection forming of a complex shaped component, Proc. 7th Int. Cold Forging Congress, 24–26th, April, Birmingham, UK, p. 79, 1985.
20. Ralendra, R., Colla, D. and Petersen, S.B., Simulation of preform design for injection forging of tubes, Proc. of the 11th NCMR, Leicester, U.K, p. 82, 1995.
21. Ralendra, R., Petersen, S. B. and Colla, D., Preforming considerations for the injection forging of tubes, IMechE Conference Transaction 1995–3, p. 79, 1995.
22. Pipan, J., Radial extrusion combined with inside ironing of tube, UMFORMTECHNIK, 27/4, p. 262, 1993.
23. Filho, L. A. M. and Al-Qureshi, H. A., Unconventional tee forming on metal tubes, *Trans. ASME, J. Engg. Ind.*, 107:392, 1985.
24. Thiruvarudchelvan, S., Elastomers in metal forming: a review, *J. of Mater. Process. Technol.,* 39:55, 1993.
25. Qin, Y. and Balendra, R., Simulation of the forming sequence of pressure-assisted Injection Forging of thick-walled tubes, TRANSACTIONS of NAMRI/SME, Vol. XXIII, p. 27, 1995.
26. Qin, Y. and Balendra, R., Technical evaluation of pressure-assisted Injection Forging of thick-walled tubes, *Advances in Manufact. Technol.* IX, Taylor & Francis, London, p. 62, 1995.
27. Balendra, R. and Qin, Y., Feasibility and technical limitation of pressure-assisted Injection Forging of thick-walled tubes, Proc. of 5th ICTP in Ohio, 7–10th October, USA, p. 327, 1996.
28. Geiger, R., From near-net-shape to net-shape cold forging—State of the art, Proc. of the 9th Cold Forging Congress, 22-26 May, UK, p. 59, 1995.
29. Nakano, T., Modern applications of complex forging and multi-action forming in cold forging, *J. of Mater. Process. Technol.,* 46:201, 1994.
30. Lange, K., *Handbook of metal forming,* Mcgraw-Hill, New York, 1985.
31. Keller, J., Optimierung der fertigungsfolge kaltmassivumformen und spanen, *Neu. Entwichlungen in der Massivumformung,* p. 8/1, 1989.
32. Dean, T. A., Progress in net-shape forging, *Adv. Techol. of Plasticity,* 2:1031, 1993.
33. Plancak, M., Bramley A., and Osman F., Non-conventional cold extrusion, *J. of Mater. Process. Technol.,* 34:465, 1992.
34. Shinozaki, K., Further investigation of cold lateral extrusion to form staggered branches from a cylindrical billet, *Annals of the CIRP* Vol.38/1/1989, p. 253, 1989.

35. Kobayashi, S., Oh, S., and Altan, T., *Metal forming and the finite-element method,* Oxford University Press.,1989.
36. Bor, C. R., Rebelo, N., Rydstad, H., and Schr der, G., *Process modelling of metal forming and thermomechanical treatment*, Springer-Verlag., Berlin,1986.
37. Tang, J., Wu, W.T., and Walters, J., Recent development and applications of finite element method in metal forming, *J. of Mater. Process. Technol.,* 46:117, 1994.
38. Cescutti, J. P., Soyris, N., Surdon, G., and Chenot, J. L., Thermo-mechanical finite element calculation of three-dimensional hot forging with remeshing, *Adv. Technol. of Plasticity,* Vol. II, p. 1051, 1990.
39. Wu, W. T., Li, G. J., and Tan, J. P., Finite Element analysis of three dimensional metal flow in cold and hot forming processes, *Annals of the CIRP,* Vol. 43/1/1994, p. 235, 1994.
40. Durham, D. R., Von Turkovich, B. F., and Assempoor, A., Modelling the frictional boundary condition in material deformation, *Annals of the CIRP* Vol. 40/1/1991, p. 235, 1991.
41. Hibbitt, Karlsson & Sorensen Inc., Contact and friction analysis with ABAQUS (User's Manual). 1991.
42. Altan, T. and Miller, R. A., Design for forming and other near net shape manufacturing processes, *Annals of the CIRP,* Vol. 39/2/1990, p. 609, 1990.
43. Osakada, K., Yang, G. B., Nakamura, T. and Mori, K., Expert system for cold forging process based on FEM simulation, *Annals of the CIRP,* Vol. 39/1/1990, p. 249, 1990.
44. Alberti, N., Cannizzaro, L., and Micari, F., Knowledge-based systems and FE simulations in metal forming processes—an integrated approach, *Annals of the CIRP,* Vol. 40/1/1991, p. 295, 1991.
45. Bakhshijooybari, M., Pillinger, I. and Dean, T. A., Development of an intelligent knowledge-based system (IKBS) for forging die design, *J. of Mater. Process. Technol.* 45 : 1-4 : 689, 1994.
46. Milner, P. R., The injection upsetting process, Leeds University, UK. 1971.
47. Balendra, R., Considerations in the modelling of injection upsetting, *Int. J. Prod. Res.,* 25:6:889, 1987.
48. Gordon, W. A., and Van Tyne, C. J., Injection upsetting—an upper bound analysis, Proc. of 15th NAMRC, 2:278, 1987.
49. Ghosh, T. P. and Tripathy, D. K., Upper-bound analysis of radial extrusion, *J. Inst. Eng. India,* 63:6:218, 1982.
50. Balendra, R. and Hijazi, M. A., Evaluation of injection upsetting models, *Int. J. Mach. Tools Manufact.*, 29:3:345, 1989.
51. Balendra, R., Process mechanics of injection upsetting, *Int. J. Mach. Tools Manufact., 2*5:1:63, 1985.
52. Qin, Y. and Balendra, R., Development of the flaws during Injection Forging, Advances in Manufact. Technol. VIII, Taylor & Francis, London, p. 587, 1994.
53. Wifi, A. S., Large strain elastoplastic analysis of some hydrostatic pressure-aided deformation processes, *Proc. of Int. Conf. Numerical Methods in Ind. Forming Processes,* 12–16, July, Swansea, UK, p. 135, 1982.
54. Andersen, B. and Andersen, C. B., Radial extrusion of tubular components (Thesis for M.Sc), Technical University of Denmark and University of Strathclyde. 1991.
55. Bay, N., Friction stress and normal stress in bulk metal-forming processes, *J. of Mech. Working Technol.,* 14:203, 1987.
56. Hibbitt, Karlsson & Sorensen Inc., ABAQUS User Manual. 1995.
57. Balendra, R. and Qin, Y., Die-elasticity errors during nett-forming of engineering components, Advances in Manufact. Technol. IX, Taylor & Francis, London, p. 204, 1995.
58. Qin, Y. and Balendra, R., FE simulation of the influence of die-elasticity on component dimensions in forward extrusion, *Int. J. of Mach. Tools Manufact.,* 37:2:183, 1997.
59. Balendra, R., and Qin, Y., Analysis and evaluation of errors in nett-form manufacturing of components, Proc. of the 3rd Int. Conf. on CIM, Singapore 11–14 July, p. 1165, 1995.
60. Lu, X. and Balendra, R., Evaluation of FE models for the calculation of die-cavity compensation, *J. of Mater. Process. Technol.*, Vol. 58/2-3, p. 212, 996.
61. Qin, Y. and Balendra, R., Limiting aspect ratios of pdz in Injection Forging, Proc. 32nd Int. MATADOR Conf. Manchester, UK, July, p. 349, 1997.

62. Schøtzle, W., Radial extrusion of flanges on steel cylindrical workpiece (in German), Technical Report 93, Institute for Metal Forming, University of Stuttgart, Springer, Berlin.1987.
63. Zhao, G., Wright, E., and Grandhi, R. V., Forging preform design with shape complexity control in simulating backward deformation, *Int. J. Mach. Tools Manufact.,* 35:9:1225, 1995.
64. Balendra, R. and Qin, Y., Decision-support-system for the comparative evaluation of processing options during component design, *Proc. of the Tenth Int. Conf. on CAPE,* 7–9 June, Palermo, Italy, p. 283, 1994.
65. Qin, Y. and Balendra, R., Geometric modelling and feature manipulation of component in an integrated multi-technology manufacturing system, Proc. of the 2nd Int. conf. on Managing Integrated Manufact., Leicester, 26–28th June, UK, p. 171, 1996.
66. Qin, Y. and Balendra, R., Computer-aided design and assessment of tools and components for nett-shape manufacturing, IMechE Conference Transaction 1995–3, p. 5, 1995.
67. Qin, Y. and Balendra, R., Principal feature modelling and its applications in the integration of design and manufacture, Proc. of the 3rd Int. Conf. on Manufact. Res. in Hong Kong, 12–16th Dec., p. 251, 1995.
68. Roy, R., Chodnikiewicz, K., and Balendra R., Interpolation of forging preform shapes using Neural Networks, *J. Mater. Process. Technol.* 45:695, 1994.
69. Spatial Technology, Inc., ACIS User Manuals. 1993.

5

Optimal Design Techniques in Non-Steady-State Metal Forming Processes

5.1 Introduction
Computer-Aided Design in Metal Forming : A New Field of Study • Presentation
5.2 State of the Art of Design Techniques for Non-Steady-State Metal Forming Proceses
Basic Design Techniques • Forging Simulation • Design Techniques Based on Artificial Intelligence • Backward Tracing Method • Design Techniques Based on Optimization Methods
5.3 The Optimization Problem
Numerical Simulation of the Forging Process • Forging Optimization Problem
5.4 Sensitivity Analysis–Optimization Method
Adjoint and Direct Methods • Discrete Direct Method • Derivatives Calculation • Gradient Based Optimization Algorithm
5.5 Examples
Shape Optimization of the Initial Billet • Shape Optimization of an Extrusion Die • Preform Shape Optimization for a Two-Step Forging Sequence • Preform Shape Design for a two-Step Forging Sequence
5.6 Conclusion
Optimal Design Techniques in Non-Steady-State Metal Forming Processes • Prospect

Lionel Fourment
Center for Forming Processes
Tudor Balan
CER ENSAM de Metz
Jean-Loup Chenot
Center for Forming Processes

5.1 Introduction

Computer-Aided Design in Metal Forming: A New Field of Study

For a person who starts studying the optimal design techniques in non-steady-state metal forming processes, it is amazing to notice that these techniques have not been much developed. In fact, the work carried out in this particular field looks poor when we compare it with the abundant literature on optimization techniques in other closely connected fields. For instance, a lot of books and reviews have been specifically dedicated to these techniques in structural mechanics, while it seems that, from a theoretical point of view, the basic equations are not very different from those arising in metal forming.

From a first standpoint, the reasons for this can probably be found in the different economic stakes which are more oriented toward mass production rather than security parts, in the internal structure of the companies and their scientific background, as these companies have moved more recently toward computer-aided techniques. From another standpoint, this delay is not amazing. In fact, non-steady-state metal forming problems are much more complex. Until the beginning of the nineties, there was no actual robust and reliable method to study them. Efficient simulation softwares have emerged only recently, making it difficult to study the design optimization problems before that time. However, this presentation would not be fair without mentioning the pioneer work of Kobayashi et al. [57], who were the very few scientists handling these issues in the eighties. So, in a certain way, for non-steady-state metal forming processes, optimal design techniques is actually a new field of study.

Presentation

The main objectives of this contribution are to first give a survey of the state of the art of the research work which have been carried out in computer aided design techniques for forming processes. Then the more promising techniques will be described, with a special focus on the gradient based optimization methods. Finally, some of the preliminary and very encouraging optimization results will be shown, along with what can be expected in a near future.

As the field of metal forming processes covers a wide domain, it would be impossible to address all the applications in a single chapter of this book. We shall then focus on a rather specific process, forging. However, it must be pointed out that forging encompasses a lot of different processes and a wide range of alloys: from unit parts to mass production, from very small and light parts to parts of several tons, from economic rough preforms to elaborated security parts. Moreover, the forging process can be considered as one of the most complex process, at least from the numerical standpoint. So, it is believed that the techniques that are described in this chapter can be easily extended to most of the other forming processes.

5.2 State of the Art of Design Techniques for Non-Steady-State Metal Forming Processes

Basic Design Techniques

The forging problem can be summarized as to find the best process design, which makes it possible to form the required part. We do not consider the preliminary problem of dressing. We assume that the forged part has already been designed out of the shape of the final part, which will often be obtained after machining. The problem is just to design a deformation path for the material in order to obtain a prescribed shape with satisfactory properties and for the lower cost. Of course, the deformation paths are not unique, and that can include several forging operations. So the problem is to find the best shapes of the preforming tools, the best forming temperatures (both the initial temperature of the billet and the temperature of the dies), the best forging velocities, the best lubricant and lubrication conditions, etc.

The optimality conditions regard several parameters which may depend on the process itself. However, most of the time, the optimal forging design has to obtain the right final part without major defects such as folds, to minimize the number of operations, to minimize the size of the initial billet, to reduce the maximum forging force during the operations, and so on.

This is a complex design problem as often the material flow is three-dimensional and difficult to foresee. It is not possible to simplify it into a less complex problem which could be more easily studied. In fact, in this area, there is a deep lack of simple mathematical models. Just a few problems can be analytically solved, such as the upsetting of a cylinder (or a bar) between flat dies without friction. Although it provides some interesting tendencies, it is far too simple to be useful for the design of a close-die forging process. So, the industrial practice was mainly based on thumb rules and empirical knowledge which have been obtained either by actual experiences, by reduced scale or simulation experiments, or by more complex mathematical models [51, 2].

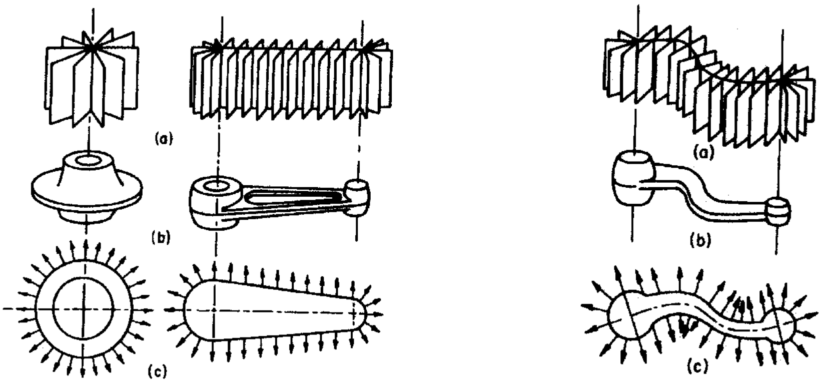


FIGURE 5.1 Decomposition of a three-dimensional forging flow into several plane strain flows which can be then studied with a simpler two-dimensional method.

Some approximated models and methods have been developed, for instance by Chamouard [11], in order to predict the filling of the forging dies and the forging force. Unfortunately, they are not easy to use. In order to bring them within the reach of these persons, recently, the Chamouard's approach has been incorporated into computer software [56]. However, according to the present computer performances and the restricting hypothesis of such models, this is probably not the most efficient and modern way to simulate the process, as we shall see.

As a matter of fact, all the proposed methods are restricted to two-dimensional (axisymmetrical or plane strain approximation) problems. For complex three-dimensional flows, the 3D problem has first to be decomposed into several 2D plane strain problems (see Fig. 5.1), which is both not easy and not always possible.

Forging Simulation

According to the difficulties which have just been presented, the design problem often results in trial and error. It is both time and money consuming, as the shapes of the dies are complex, difficult, and expensive to produce. In order to lower these costs, several simulation strategies have been developed to replace the actual trials. Easy to form model materials such as lead, plasticine [52], and wax [64] can be substituted to the current metal, making it possible to reduce the forming force. These materials, and more particularly plasticine and wax which are more widely used today, exhibit nearly the same behavior as the metal under hot isothermal conditions. One of the main restriction of this approach is the temperature dependency of the model materials, which is quite different from the behavior of the metal. However, for most of the processes, the forging is so fast that the heat exchanges with the dies is small and that the heat generated by the material deformation is not enough to significantly modify the material flow. Thus, this technique provides an easy way to study the material flow, to predict the major defects such as folds [42] and insufficient pressure in the die details, and to estimate the forging force. Moreover, it is possible to study the material deformation and the fibering by mixing model materials of several colors. With some mechanical models, the total strain can be computed out of these pieces of information [52]. By mixing model materials with different properties, or by adding some other components, the behavior of the model material can be slightly adjusted to the behavior of the studied metal. As often for metal forming processes, the friction phenomenon is difficult to simulate. However, some kind of model materials for the lubricants can be proposed.

The main shortcoming of this simulation approach is that although the dies can be produced into a less expensive material and without heat treatment, they are as complex as the actual process, which requires significant time and energy to produce them. Moreover, as has been mentioned earlier, the thermal effects cannot be taken into account.

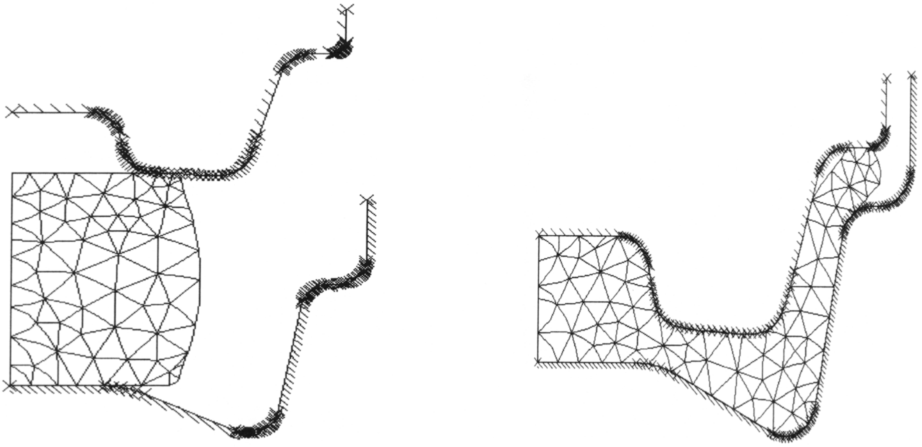


FIGURE 5.2 First forging operation of an aeronautical rotor, beginning and end.

Nowadays, the numerical simulation provides much faster results as the toolings are only virtually designed. Moreover, these results are more accurate and more varied, such as the flow at different times of the process, the velocity field, the strain, the strain rates, the stresses, the temperature, the tool wear, and the tool deformation. Several softwares have been marketed, such as the well spread FORGE2© [9], [40] and DEFORM2D© [65] (initially called ALPID), [47] for axisymmetrical and plane strain problems. They are actual tools for designers and their industrial use is ever increasing. Indeed, they make it easy to quickly find the main shortcomings of the studied design, and then to test several modifications to improve it. For expensive safety parts, or for very large single parts, they also provide a quality insurance as they give an estimation of the mechanical characteristics of the part which should otherwise be obtained by destructive testing. Regarding the true three-dimensional problems, automatic remeshing difficulties, as well as computational time and memory requirements, have long hindered the industrial use of these softwares. Nowadays, both the software and hardware progresses have made these computations possible. They are used in forging companies, for instance to understand the development of a fold and test whether a new preform design can remove it [22]. The tool shape discretization required for the numerical simulation, often a finite element mesh of the tool surface by triangles, can almost directly be obtained from the CAD design. It reduces to rather insignificant times the specific numerical design required by the finite element simulation. FORGE3© [13], [15] and DEFORM3D© [66] are such available softwares.

The following example shows how the numerical simulation can be used to design a forging sequence. It has been carried out using the FORGE2® software. The problem is the forging of an axisymmetrical aeronautical rotor in three steps, starting from a cylinder. The preliminary upsetting (Fig. 5.7) only provides the correct high/radius ratio, so it does not require a specific study. The first actual forging step, the preforming operation (Fig. 5.2) has to be optimized. In fact, during the second and finishing operation (Fig. 5.3), as in the actual process, a piping defect is observed (Fig. 5.4) under the inside rib. The study of the velocity field (Fig. 5.5) helps to understand the defect formation, due to the conjunction of two flows to fill the rib. After several trials and errors, a better preform shape is proposed to prevent this kind of flow. Figure 5.6 shows the modified upper tool of the preforming operation, and the resulting flow during the finishing operation. It can be noticed that the conjunction of flows has been removed. Starting from this new preform, the final shape is well obtained, without noticeable defects. The process is then more precisely studied, with a special focus on the material deformation. Marks are made on the initial billet (Fig. 5.7). Their deformations can be observed at the beginning and end of the preforming operation (Fig 5.8), at an intermediate step of finishing operation with a zoom in the previously folded zone, and

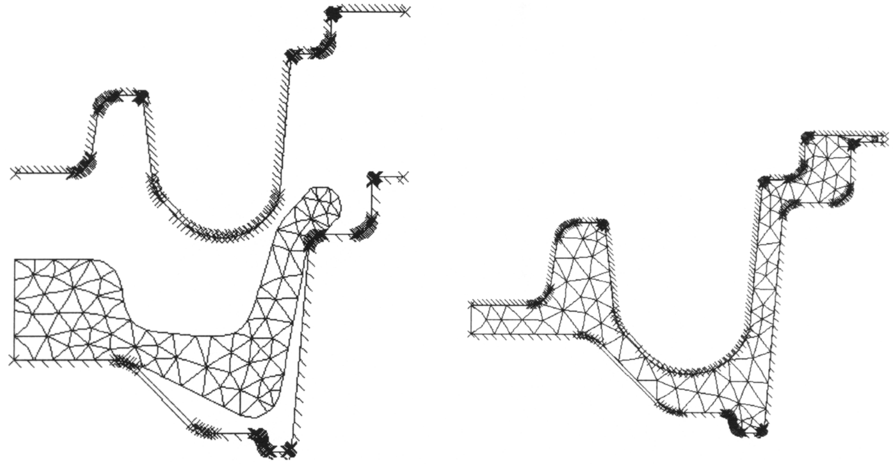


FIGURE 5.3 Second forging operation, beginning and end.

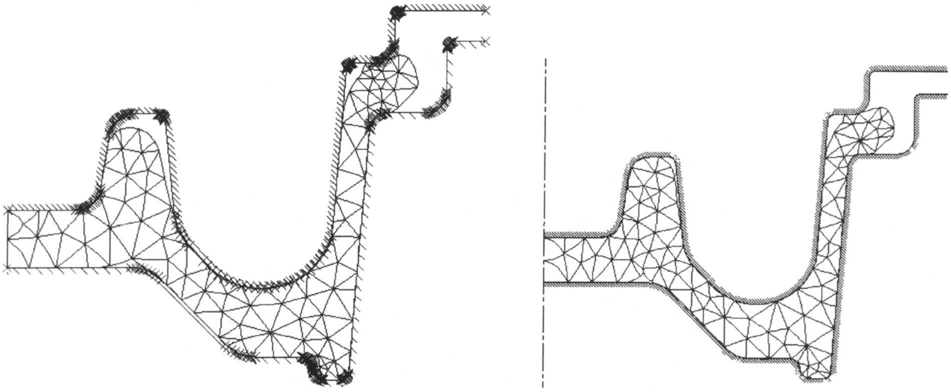


FIGURE 5.4 Two steps of the formation of the fold.

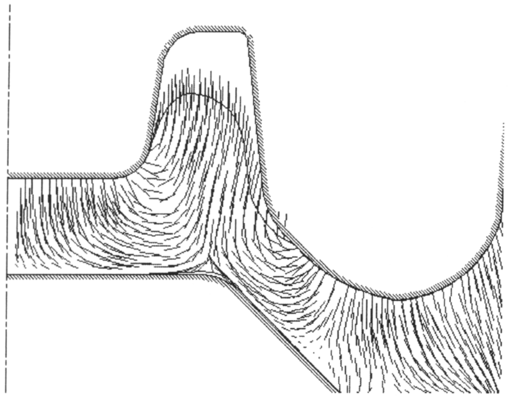


FIGURE 5.5 Velocity field at an intermediate step of the second forging operation—the piping defect is clearly explained by two concurrent flows which fill the upper rib.

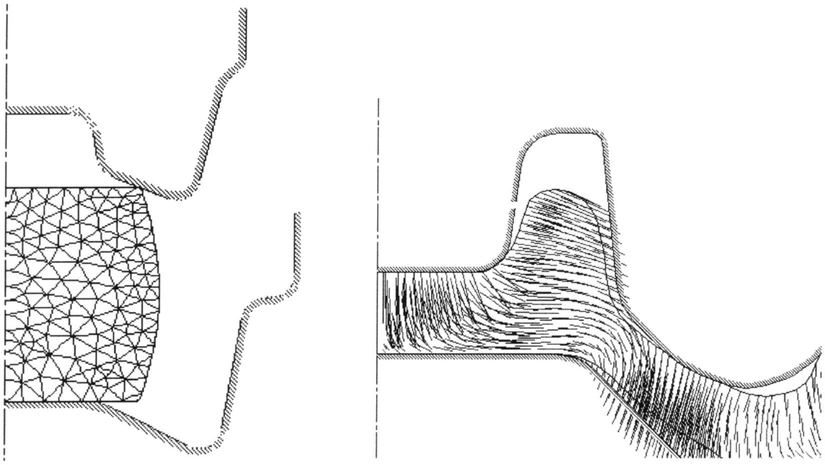


FIGURE 5.6 New design of the upper die of the preforming operation and the resulting flow in the piping zone during the finishing operation.

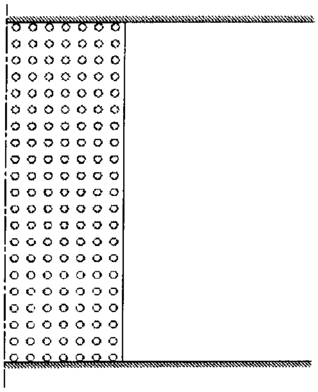


FIGURE 5.7 Optimized forging sequence: preliminary upsetting operation, marks drawn on the initial workpiece.

at the end of the process (Fig. 5.9). It shows that the material is not homogeneously deformed in the previously folded zone, and that further optimizations could be carried out.

So, the numerical simulation provides a powerful tool both to test a forging design and to get a better understanding of the material deformation in order to possibly improve it. Although it does not provide any designing strategy, this tool makes up the basis for the optimal design techniques.

Design Techniques Based on Artificial Intelligence

In a first time, it seems natural to use the Expert Systems technique to solve the present design problem. One the one hand, it can provide a fast initial analysis of the problem, and suggest a first design [Len94]. On the other hand, it makes it possible to keep the expertise of experienced people, which is stored both in the data base of the system and in the set of rules. Finally, as the other optimization techniques described in the following, it helps in defining standard design rules and so provides a consistent methodology throughout a company. Thus expert systems have been developed for forging problems [4, 38]. However, it seems that this approach faces some basic difficulties. First, the human expertise is difficult to gather, because it is difficult to clearly describe a thought process, and because the vocabulary used is not always well defined. The same phenomena can be described with different words, in different companies and even sometimes inside the same company. Some persons may also consider a certain

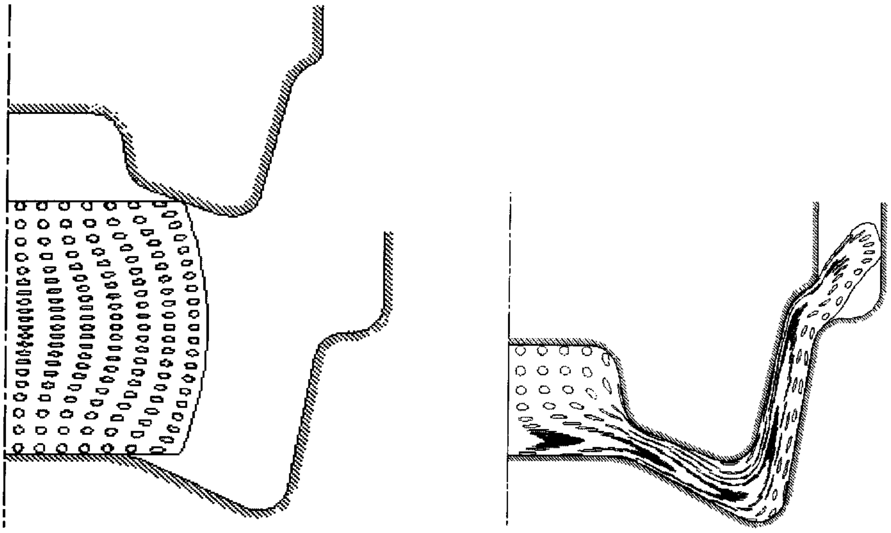


FIGURE 5.8 Optimized forging sequence with marks: preforming operation, beginning and end.

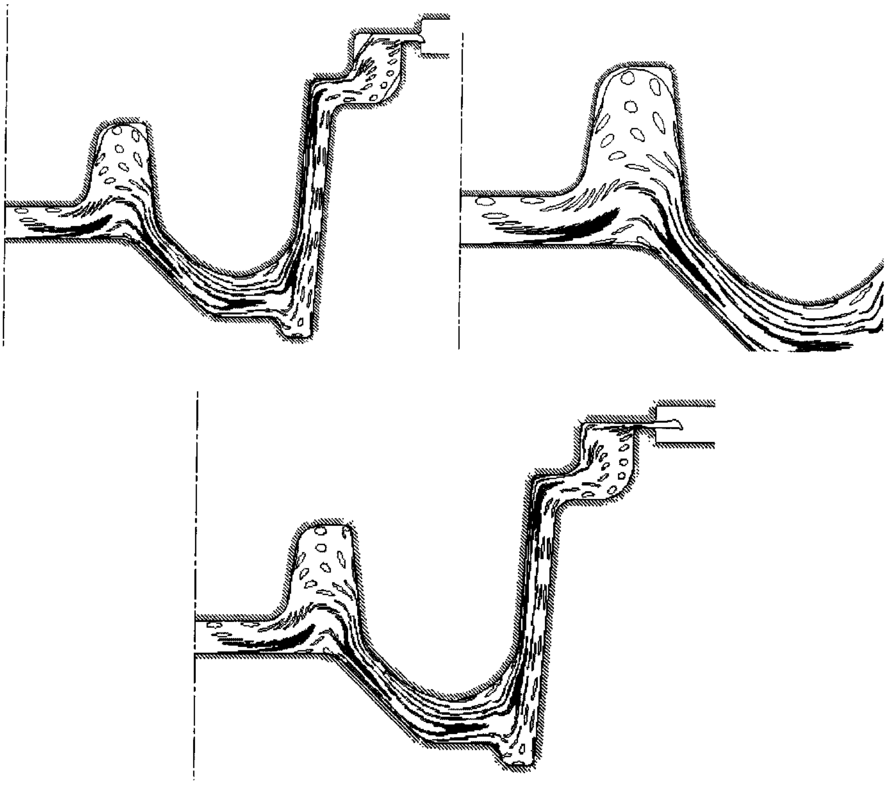


FIGURE 5.9 Optimized forging sequence with marks: an intermediate step of the last forging operation with a zoom in the previously folded area and the end of the last forging operation.

phenomenon as a defect, while others may not because they do not have the same focus. Moreover, the other usual shortcomings of expert systems are also faced in metal forming : the difficulty to verify the expertise and to check the integrity of different sources of expertise, in order to avoid antinomies, and the fact that expert systems cannot give an answer if the expert does not know it. Therefore, they allow non expert people to study simple designs without requiring the presence of an expert, but they cannot increase the knowledge.

In order to eliminate some of these shortcomings, the Case Based Reasoning techniques have been developed, in particular when the rules are not easy to define. They are based on the analogies between the different problems. First, they provide another software environment to store all the experimental design results, so allowing, as expert systems, to keep the know-how of a company. Then, when a new design has to be defined, the system computes the distance between the new problem and the already solved problems stored in the data base. This way, it can suggest one or several solutions close to the user's problem. This is just a first solution as well as indications which will eventually have to be improved. However, if the design problem does not belong to one of the family of the database, the system will require a further help from the expert. This approach has been successfully applied to forming processes [67]. However, on one hand, it is limited to a specific process, which means that there are different softwares for different families of parts, and on the other hand, it does not actually solve the design problem.

Consequently, these two approaches can essentially be used to reduce the number of initial trials, but the suggested design still has to be validated by calculations or experiences. However, they do not really increase the knowledge about the design process and the design techniques. The other approaches, which are described in the next sections, seem more promising.

The numerical simulation can efficiently be used to solve the various problems which have just been raised: gathering the human expertise, standardizing the pieces of information and results, and verifying the outputs of the artificial intelligence softwares. Actually, it can easily train the system. However, in this case, the neural network, genetic algorithm approaches seem better adapted. Moreover, using the biological system analogy, they make it possible to suggest a design for a rather new problem. The optimization methods provide another type of design strategy which can also suggest innovative designs.

Backward Tracing Method

In the early 1980s, a pioneer work was carried out on the basis of the finite element simulation of the forming processes [57]. In the frame of numerical based approaches, it is very specific as it is a constructive method for preform design in forging or other related processes. This way, it can provide new and original designs. The basic idea is to trace back the deformation path. The procedure starts from the final desired part and computes a backward deformation path until the beginning of the process. It so proposes a preform which will make it possible to obtain the final part. Thus, this method consists in winding up time, which is a very complex problem. At each time increment, $t + \Delta t$, upon the $\Omega^{t+\Delta t}$ configuration (where Ω represents the domain of the workpiece), the algorithm computes the Ω^{t} previous configuration, the corresponding velocity field V^{t}, and contact conditions such that they would provide the $\Omega^{t+\Delta t}$ configuration if the direct calculation was carried out. However, it is clear that the backward deformation path is not unique and that the contact phenomenon is irreversible. Several backward algorithms have been investigated for releasing the contact nodes, either using a prescribed sequence or a specified criterion, or by recording the boundary changes during the forward simulation. So, the method significantly relies on the selected strategy, which makes it quite difficult to produce a general and robust software.

This method has been successfully applied to preform design for several forming 2D and 3D processes, such as plane strain rolling, [43], three-dimensional rolling [44], and axisymmetrical forging [41, 68] (see Fig. 5.10).

When the material has a complex behavior, for example when it exhibits a dependency on the strains and temperature, this approach faces some additional difficulties regarding the irreversibility of the deformation path. However, some applications have been carried out in these cases.

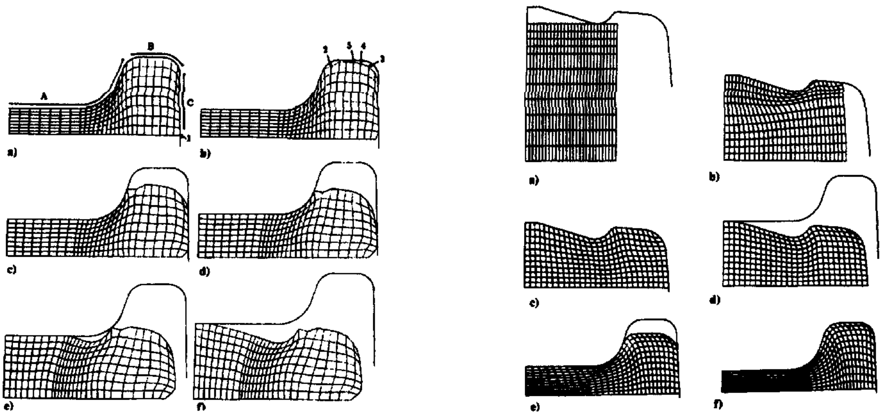


FIGURE 5.10 Backward tracing method for the preform design of an axisymmetrical part—left: backward simulation starting from the final part—right: forward simulation using the preforming tool derived from the obtained prefom (a-c) preforming operation (d-e) finishing operation [41] (with permission).

This powerful method has some important shortcomings which have motivated some of its users to move to more general optimization methods [69]. First, it does not actually suggest a design of the preforming tools, but the shape of the preform itself. So there is no guarantee that this preform can be produced by forging, and that the preforming tools might not be too complex. In fact, it seems difficult to introduce shape constraints for the preforming tools. Second, it is not possible to know whether the proposed preform is optimal or not with respect to the design objectives. This way, this method can be rather considered as an efficient way to suggest new preform shapes, which will be further investigated and optimized by finite element direct simulations. The concurrent optimization methods, although they look less advanced today, have much more potential.

Design Techniques Based on Optimization Methods

These methods are based on the direct numerical simulation of the process. Using optimization techniques, several designs are tested in order to find the best one. In a first step, it is necessary to identify the various parameters, $(p_k)_{k=1,\mathrm{Npar}}$, of the process which have to be optimized. Npar is the number of such parameters and $\boldsymbol{p} = (p_k)_{k=1,\mathrm{Npar}}$ is the vector of the parameter values. As mentioned earlier, the possible parameters are: the shape of the initial billet, the shape of the dies of the preforming tools, the process parameters such as the dies velocities and temperatures, the billet temperature, etc. In a second step, a mathematical function, the objective function Φ, has to be defined in order to measure the quality of the design for a given set of parameters. For instance, Φ can be the distance between the obtained and the desired part, the severity of the forming defects such as folds, the forming energy, the metallurgical qualities of the final part, etc. Eventually, the design has to obey a certain number of constraints. There can be both equality C_1 and inequality C_2 constraints, such as geometrical constraints regarding the shape of the preforming dies, the maximal values of the forming velocities and pressures, or any of the fundamental quality requirements such as the absence of defects. So, the problem can be summarized as:

$$\text{find } p \text{ such that: } \Phi(\boldsymbol{p}) = \underset{\boldsymbol{p}',\ \begin{cases} C_1(\boldsymbol{p}') = 0 \\ C_2(\boldsymbol{p}') \leq 0 \end{cases}}{\mathrm{MIN}} \Phi(\boldsymbol{p}') \tag{5.1}$$

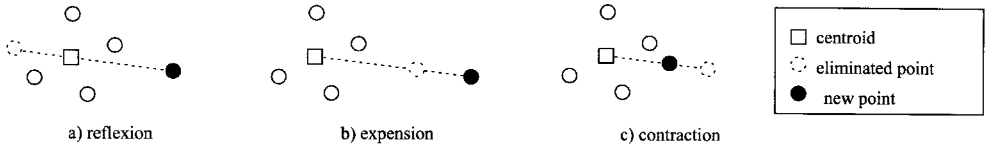


FIGURE 5.11 The different strategies of the simplex algorithm for the selection of a new point.

The numerical simulation is used to compute the different values of Φ and C. Formulated, this problem is similar to the optimization problems met in structural mechanics. Several optimization algorithm can be used to solve it.

Simplex Optimization Algorithm

This algorithm is based on a preliminary definition of a set of possible optimal points. At each iteration, a new point is added while the worst point is removed [39]. In a first step, the objective function is evaluated for a prescribed number of points, belonging to the space of the parameters. The number of points must be greater than the number of parameters. During the iterations of the algorithm, the point which provides the larger value of the objective function is removed. It is replaced by a new point, derived from its value, whose position is computed using the following: first reflection with respect to the gravity center of the set of points, then expansion or contraction according to the value of the objective function at this point (see Fig. 5.11).

This method has been applied to the shape optimization of an extrusion die profile in steady-state forming, in order to minimize the extrusion force [49]. Although it requires a large number of function evaluations, in this specific two-dimensional steady state example, the algorithm shows as efficient as a gradient based method.

Genetic Optimization Algorithm

Here also, the method is based on a set of possible points, which is used to generate new solutions. However, the algorithm is quite different, as it is derived from the biological evolution theory, where the best specimens of a population are used to generate a new and better generation of points. Here, better refers to the value of the objective function. Reproduction, crossing, and mutations rules govern the generation of the new set of points [24], [33]. This kind of probabilistic algorithm makes it possible to obtain a local minimum of the objective function, even when the function is not continuous nor differentiable. This is its greater power. On the other hand, they require a large number of function evaluations. For instance, some version of the algorithm using a small population needs about 200 function calculations to find an optimum [39], [48], which is too many for complex forging problems.

It has been applied to the design of an industrial four steps forging sequence, in order to minimize a damage criterion [21]. The process parameters are the angles of the different preforming dies, the radius/length ratio of the initial billet, and the number of forging operations. Although this last parameter takes integer values and so is discontinuous, the genetic algorithm can handle it. For the next more efficient methods, the continuities of the Φ, C functions, and p are the basic requirements.

Analytical Approximation of the Objective Function

Another type of optimization algorithm is based on a preliminary approximation of the objective function, which can then be easily minimized using any efficient algorithms.

As for the simplex and genetic algorithms, an initial set of parameters values $(\boldsymbol{p}_k)_{k=1,\mathrm{Nbset}}$ is first defined. This set must be large enough in order to represent all the possible variations of the parameters. It can be designed using the plane of experiments methods. The objective and constraint functions, $\Phi(\mathbf{p}_k)$ and $\mathrm{C}(\boldsymbol{p}_k)$, are evaluated for all these parameter values. If Nbset, the number of $\boldsymbol{p}_k$ points, is sufficiently larger than the number of parameters, N_{par}, it is possible to build approximations $\tilde{\Phi}$ and $\tilde{C}$, respectively of Φ and $\tilde{C}$. Different interpolation functions can be used, such as splines or polynomials [50, 60]. Neural networks can also be used [45]. In this case, the network is trained during the first stage of the method and then used for the optimization stage. So, the complex optimization problem (1) is replaced by the approximated problem:

$$\text{find } p \text{ such that: } \tilde{\Phi}(\tilde{\boldsymbol{p}}) = \underset{\boldsymbol{p}',\ \begin{cases} C_1(\boldsymbol{p}') = 0 \\ \tilde{C}_2(\boldsymbol{p}') \le 0 \end{cases}}{\text{MIN}} \tilde{\Phi}(\boldsymbol{p}') \tag{5.2}$$

With the analytical interpolation functions, gradients and second order gradients of $\tilde{\Phi}$ and $\tilde{C}$ can be easily calculated. So the efficient BFGS or Newton-Raphson algorithms can be used to find p. The evaluations of $\tilde{\Phi}$ and $\tilde{C}$ being immediate, any kind of algorithm can be used, and the total time of the minimization is negligible with respect to a single evaluation of $\Phi(\boldsymbol{p}_k)$. Then $\Phi(\tilde{\boldsymbol{p}})$ can be computed and compared to $\tilde{\Phi}\ (\tilde{\boldsymbol{p}})$. If $\Phi(\tilde{\boldsymbol{p}})$ is not satisfactory, then $\tilde{\boldsymbol{p}}$ can be added to the set $(\boldsymbol{p}_k')_{k = 1,\text{Nbset}}$ and the approximation of $\tilde{\Phi}$ can be improved using this value. If the neural network strategy is used, the network is further trained with this value. Or else, a new set of $(\boldsymbol{p}_{k'})_{k'=1,\text{ Nbset}}$, parameter values can be generated in the neighborhood of $\tilde{\boldsymbol{p}}$, in order to provide a better new $\tilde{\Phi}$ approximation near the expected optimal value.

This type of method is well adapted when the number of parameters is not large, or when the parameters have a small range of variations. Indeed, it requires at least about 2^{Npar} simulations of the process. The required computational time is the main limitation of this easy to implement method. On the other hand, the parallel computers or the parallel networks of sequential computers provide a straightforward speed up of this type of easy to parallelize method. For non steady-state forming processes, it is more reliable that the following finite difference based methods. On the another hand, its convergence and accuracy also depend on the numerical accuracy of the calculations of the Φ functions. It has been used, for instance, to optimize the size of the initial billet in order to fill the dies at the end of forging [45] and to improve an already existing forging sequence [50, 60] for which the variation range of the shape parameters was sufficiently small.

Optimization Methods Based on the Gradient Calculations

The other way to improve the convergence rate of the optimization methods is to compute the first order gradients of Φ and C, and eventually the second order gradients. For instance, for the rather similar problem of parameter identification using the inverse method, it has been noticed that the analytical gradient calculations dramatically decrease the computational time. In fact, it allows to use faster minimization algorithms, such as the well-known DFP, BFGS, or Gauss Newton algorithms [20].

Finite Difference Gradients

For non-steady-state forming processes, first order gradients are quite complex to compute. They can be approximated by the following expression:

$$\frac{d\Phi}{dp_i}(p) = \frac{\Phi(p + \Delta p_i) - \Phi(p)}{\Delta p} \tag{5.4}$$

or

$$\frac{d\Phi}{dp_i}(p) = \frac{\Phi(p + \Delta p_i) - \Phi(p - \Delta p_i)}{2\Delta p} \tag{5.5}$$

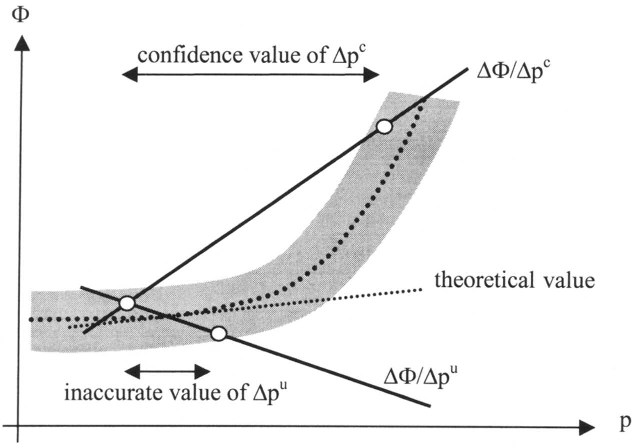


FIGURE 5.12 Error on the finite difference derivatives resulting from the discretization error on the objective function.

where Δp_i is a vector of parameter values which is almost equal to zero, but the i^{th} coordinate,

$$\text{which is equal to } \Delta p : \Delta p_i = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta p(i^{\text{th}} \text{ coordinate}) \\ 0 \\ 0 \end{bmatrix}$$

This is an easy way to compute the function derivatives, but it has two major disadvantages: it is quite time consuming as it requires N_{par} or $2N_{\text{par}}$ additional simulations of the process, and it is difficult to control the accuracy of the derivatives. In fact, it is all the better that Δp is smaller. However, the round off and discretization errors have to be taken into account, so Δp cannot get a smaller value than the accuracy of the simulation. If Δp is too small, as shown in Fig. 5.12, the difference between $\Phi(\text{p}+\Delta\text{pi})$ and $\Phi(p)$ belongs to the discretization error so the numerical derivative can get any value. Indeed, Δp must be sufficiently large, as shown in Fig. 5.12, but this also reduces the accuracy of the derivative. Although this method has been used in the beginnings, for instance for the preform shape optimization in forging [Bec89], the number of required simulations for reaching an optimal solution was large (more than 250), probably due to the poor accuracy of the numerical derivatives.

On the other hand, for more complex forging processes which require several remeshings, the discretization error is even increased during the transfer of domain and variables from one mesh onto the other. It can get so large that the numerical derivatives do not make sense any longer. The following simple example shows it. This is the forging of a simple cylinder by a die, the shape of which is discretized by a three points spline (see Fig. 5.13). Three remeshings are required to compute the prescribed deformation.

The time evolution of the derivative of the energy objective function is shown in Fig. 5.14. This derivative has been computed both analytically, as it will be described in the next sections, and using the finite difference method. Two Δp values, 10% and 100%, have been tested. With $\Delta p = 10\%$, after the first remeshing the numerical derivative is just slightly different from the analytical value, while after the third remeshing, the numerical derivatives oscillate so much that they do not make sense. With $\Delta p = 100\%$, the oscillations disappear, but the accuracy of the derivatives is poor with respect to the analytical values, as shown in Fig. 5.14.

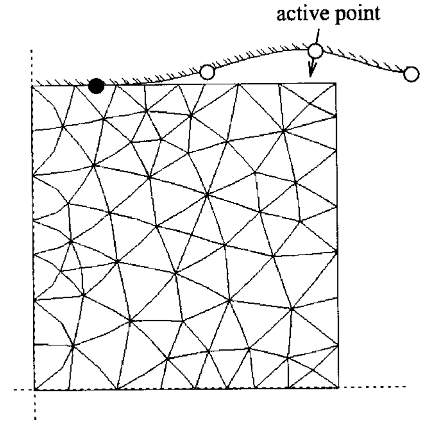


FIGURE 5.13 Test problem—forging of a cylinder using a die defined by a four points cubic spline—the arrow shows the perturbed characteristic point.

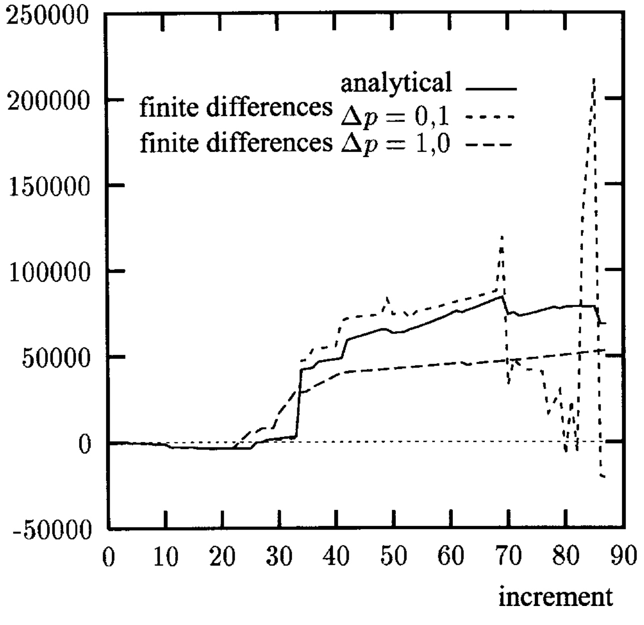


FIGURE 5.14 Numerical and analytical derivatives of the energy objective function versus the increment number.

This shows that, for standard non steady state forming problems, numerical derivatives cannot be used and that an effort has to be made to compute them analytically.

Introduction to the Gradient Calculation

In order to carry out the derivatives calculations, which is also called the *sensitivity analysis*, several methods can be considered. They can be split in two main classes: the methods based on the direct differentiation and the methods based on the adjoint state. These two approaches are developed and compared later. However it is still possible to split each of these methods into two other classes, whether the differentiation is first carried out at the discrete or at the continuous level.

Variational Differentiation Methods

Following the variational approaches, the problem equations are differentiated before being discretized. From a mathematical standpoint, it could probably be demonstrated that these two operations are symmetric, or almost symmetric, as numerical experiments show. However, from a strict numerical standpoint, the symmetry is not obtained due to the round off errors. According to this, the variational approaches are sometimes considered to be less consistent, as the resolutions of the two problems, the direct forging and the sensitivity analysis, are not based on the same discretization scheme. This is rather unavoidable when the sensitivity analysis is carried out independently, for instance, when the source code of the direct problem is not available. On the other hand, these approaches make it possible to extend the mathematical developments, so gaining the favors for a more fundamental approach. In the frame of fluid mechanics, they are rather popular and their extension to metal forming problems has been studied [36].

Generally, the objective function can be written as an integral of a function of various state variables. For example, the forming energy is such a function:

$$\Phi = \int_{t=0}^{t=t_{\text{end}}} \int_{\Omega_t} \sigma : \dot{\varepsilon}\, dw dt \tag{5.6}$$

(see Section 5.3 for the problem notations). The differentiation of Φ with respect to the shape parameters $\boldsymbol{p}$ gives:

$$\frac{d\Phi}{dp} = \int_{t=0}^{t=t_{\text{end}}} \int_{\Omega_t} \left(\frac{\partial \sigma}{\partial p} : \dot{\varepsilon} + \sigma : \frac{\partial}{\partial p} \dot{\varepsilon} \right) dw dt + \int_{t=0}^{t=t_{\text{end}}} \int_{\partial\Omega} \sigma : \dot{\varepsilon} v^s \cdot n ds dt \tag{5.7}$$

where v^s is the displacement velocity of the boundary $\partial\Omega$, due to the parameter variations.

When a flow formulation is used to solve the viscoplastic forging problem, all the state variables can be expressed as functions of the velocity field, $\boldsymbol{v}$. This way, all the shape derivatives of Eq. (5.7) can be easily calculated once $\partial v/\partial p$ is known.

In order to achieve this, the problem equations are differentiated with respect to $\boldsymbol{p}$: the equilibrium and constitutive equations, and the corresponding boundary conditions. This leads to a linear system of equations, which is solved using a variational approach and a finite element discretization [Pir84]. However, when it is applied to a non-steady-state forming problem, this formulation exhibits boundary integrals on $\partial\Omega$ which comprise gradients of the stress tensor [37]. This poses a difficult problem because these gradients are not straight results of the finite element resolution: they are either unknown or calculated with poor accuracy, which results in a significant loss of accuracy in the sensitivity analysis.

In order to overpass these problems, a slightly different method has been developed [5], [6]. It is based on a total lagrangian formulation, rather than on an updated eulerian formulation. The total deformation gradient $\boldsymbol{F}$ is the main variable, rather than the velocity field $\boldsymbol{v}$. The equilibrium equation is expressed in the reference configuration Ω_0, using the Piola Kirchoff stress measure:

$$\sigma_{PK} = \det(F) \sigma F^{-T} \tag{5.8}$$

$$\forall x \in \Omega_0, \quad \forall t \in [0, t_{\text{end}}], \quad \operatorname{div}(\sigma_{PK}) = 0 \tag{5.9}$$

This local equilibrium equation is differentiated with respect to the shape parameters $\boldsymbol{p}$, and the weak variational form of the problem is written, still in the reference configuration:

$$\forall u^*, \quad \int_{\Omega_0} \left(\operatorname{div} \left(\frac{d\sigma_{PK}}{dp} \right) \cdot u^* \right) dw_0 = 0 \tag{5.10}$$

where $u*$ are the test functions for the sensitivity problem.

After integration by parts:

$$\int_{\Omega_0} \int \frac{d\sigma_{PK}}{dp} \cdot \frac{\partial u^*}{\partial x_0} dw_0 = \int_{\partial\Omega_0} \frac{d\sigma_{PK}}{dp} n_0 \cdot u^* ds_0 \tag{5.11}$$

As the reference configuration does not depend on the parameter values, the right hand term is easily written onto the current configuration. Then, the constitutive equation is differentiated so that $d\sigma_{PK}/dp$ can be expressed as a function of the derivatives of the deformation gradient dF/dp. Thus, Eq. (5.11) is

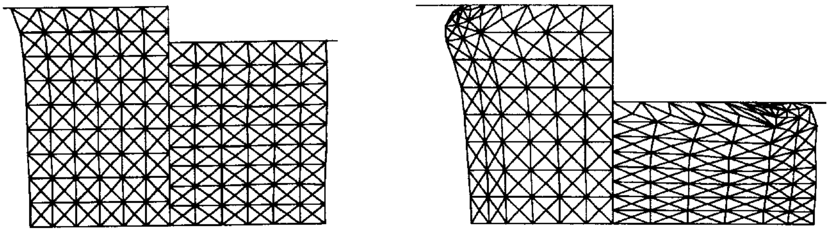


FIGURE 5.15 Optimal preform and subsequent forged part for the 17% upsetting problem. Optimal preform and subsequent forged part for the 44% upsetting problem [5] (with permission).

like an equation of the principle of virtual work, the unknown being the shape sensitivity of ***F***. For its resolution, the same finite element and time discretizations are used, as for the direct analysis. However, it must be noticed that the subsequent linear system does not exhibit the same stiffness matrix as the direct problem, while this is the case with the discrete differentiation methods.

This method has been successfully applied to the shape optimization of extrusion dies [6] and of simple forging preforms [5]. This last application consists of finding the shape of the preform that eliminates the bulging of the part after upsetting (a similar problem is presented in Section 5.5). Figure 5.15 shows the optimal preforms for two different upsetting heights. The shape is described with quintic splines and the material is assumed hyperelastic viscoplastic. For this example, the method shows the same level of accuracy as the discrete differentiation method, although it is slightly less efficient since the stiffness matrix of the sensitivity problem has to be recomputed.

5.3 The Optimization Problem

Numerical Simulation of the Forging Process

The forthcoming description will be limited to the two-dimensional axisymmetrical forging problem. In the introduction, we have justified the restriction to the forging process, and it seem that until now the design optimization techniques have not been applied to three-dimensional non-steady-state metal forming problems. Actually, the 3D simulation is still a difficult issue and the reliability and robustness of the software is very recent. We shall then restrict ourselves to axisymmetrical forging. Moreover, for the sake of simplicity, we shall describe only the hot forging problem under isothermal conditions in order to manipulate rather simple constitutive and friction equations. The isothermal hypothesis is well verified in standard forging processes. In fact, the forging time is too short to allow a significant heat exchange with the dies, and the heat generated by the deformation is not enough to actually modify the flow. The thermal calculations are necessary when thermal phenomena themselves have to be controlled during the process. The interested reader will find more details of the 2D simulation in [34, 40, 47, 47], and [27]. As for the 3D simulation, recent descriptions can be found in [12, 15], and [66].

Problem Equations

In most of the forming processes, elasticity effects are negligible, so the material is considered to be rigid plastic and incompressible. The material is assumed to be homogeneous and follows the viscoplastic Norton-Hoff law.

$$\operatorname{div}(v) = 0 \tag{5.12}$$

$$s = 2K(\sqrt{3}\dot{\bar{\varepsilon}})^{m-1}\dot{\varepsilon} \tag{5.13}$$

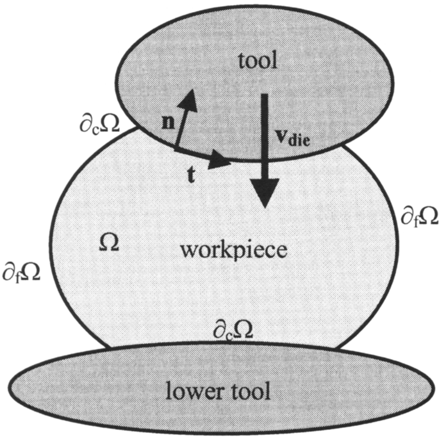


FIGURE 5.16 The forging problem.

where:

v is the velocity field
$\dot{\varepsilon}$ is the strain rate tensor, $\dot{\varepsilon} = 1/2\ (\mathrm{grad}(v) + \mathrm{grad}^t(v))$
$\dot{\bar{\varepsilon}}$ is the equivalent strain rate: $\dot{\bar{\varepsilon}} = \sqrt{2/(3)\dot{\varepsilon}:\dot{\varepsilon}}$ (5.15)
σ is the Cauchy stress tensor
s is the deviatoric part of σ, $\sigma = s - \lambda I$
and λ is the hydrostatic pressure, $\lambda = -1/3\ \mathrm{trace}(\sigma)$
K is the consistency of the material
m is the strain rate sensitivity coefficient

At the interface between the tools and the part, $\partial_c\Omega$, as the contact pressure is often high, the friction is assumed to follow a Norton law:

$$\tau = -\alpha\ K\|\ \Delta v_t\ \|^{q-1}\Delta v_t \tag{5.16}$$

where:

τ is the shear stress: $\tau = \sigma n\ .\ t$
n and t are the normal and tangent to the tool surface (see Fig. 5.16)
Δv_t is the relative tangential velocity: $\Delta v_t = [(v - v_{\mathrm{die}}).t]t$ (5.17)
v^{die} is the prescribed velocity of the die
α and q are the friction law coefficients

At any time in the process, the forged workpiece Ω is in equilibrium between the forging dies, the $\partial_c\Omega$ part of the surface, while $\partial_f\Omega$ represents the part of the free surface (see Fig. 5.16). Under standard forging conditions, inertia effects and gravity forces are negligible, so the balance equations are:

$$\mathrm{div}(\sigma) = 0 \quad \mathrm{on}\,\Omega \tag{5.18}$$

$$(V - V_{\mathrm{die}})\cdot n\ = 0 \quad \text{and} \quad \tau = -\alpha K\|\ \Delta v_t\ \|^{q-1}\Delta v_t, \quad \text{on } \partial_c\Omega \tag{5.19}$$

$$\sigma n = 0 \quad \text{on } \partial_f\Omega \tag{5.20}$$

Variational Formulation and Finite Element Discretization

In order to solve Eqs. (5.18) and (5.12), they are written again in a variational form, which is almost equivalent to the virtual work principle. In the standard way for incompressible flow problems, a velocity/pressure formulation is used. The set of kinetically admissible velocity fields (which satisfy the contact conditions Eq. (5.19)) is the set of 0 kinetically admissible velocity fields (which satisfy the homogeneous contact conditions), and the set of pressure fields.

$$\begin{cases} \forall v^* \in 0, \int\limits_{\Omega} s:\dot{\varepsilon}(v^*)dw - \int\limits_{\Omega} \lambda \operatorname{div}(v^*)dw - \int\limits_{\partial_c\Omega} \tau . v^* ds = 0 \\ \forall \lambda^* \in, \int\limits_{\Omega} \lambda * \operatorname{div}(v)dw = 0 \end{cases} \tag{5.21}$$

The finite element discretization is then introduced. As the mesh is severely distorted during the forming simulation, frequent remeshings have to take place. It requires the use of a robust automatic mesh generator, which is more common for triangles. For this reason, triangular elements are used. In order to satisfy the compatibility condition between the velocity and pressure interpolations [7, 10], the coordinates and the velocities are interpolated using quadratic interpolation functions, while the pressure is interpolated with piecewise functions (see Fig. 5.17).

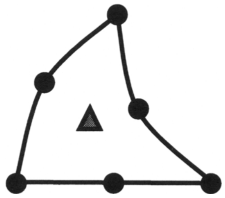


FIGURE 5.17 Triangular finite element: P2P element (quadratic velocity and constant pressure).

$$\begin{cases} X_h = \sum\limits_k X^k N^k \\ V_h = \sum\limits_k V^k N^k \\ \lambda_h = \sum\limits_l \Lambda^l N^l \end{cases} \tag{5.22}$$

Here, it is convenient to introduce the B^k operator, which is a three components tensor for a given k value, and which is defined according to:

$$\dot{\varepsilon}_h = \sum\limits_k V^k B^k \tag{5.23}$$

After discretization, and using the Galerkin formulation : $v_h^* = N^k$, and $\lambda_h^* = N^1$, where we use a standard abuse of notations, as here N^k is a vector of coordinates

$$\begin{cases} \forall k, \int\limits_{\Omega} 2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-1} \dot{\varepsilon}_h : B^K dw_h - \int\limits_{\Omega} \lambda_h \operatorname{div}(v_h^*)(dw) + \int\limits_{\partial_c\Omega} \alpha K \| \Delta v_{ht} \|^{q-1} \Delta v_{ht} . N^k ds_h = 0 \\ \forall k \in \partial_c\Omega (V^k - V_{\text{die}}).n = 0 \\ \forall 1, \int\limits_{\Omega} N^1 \operatorname{div}(v_h) dw_h = 0 \end{cases} \tag{5.24}$$

As N are piecewise functions, the last equation can also be written:

$$\text{for any mesh element, } 1, \int\limits_{\Omega_1} \operatorname{div}(v_h) dw_h = 0 \tag{5.25}$$

In practice, in order to reduce the number of unknowns and obtain a pure velocity formulation, a penalty formulation is often used. So, the following penalty term is substituted to the pressure term of Eq. (5.24):

$$\forall k, \int_{\Omega}\left(2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-1}\dot{\varepsilon}_h{:}B^k(dw_h) + \rho_{\text{pen}}\sum_{1}\left(\int_{\Omega_1} dw_h\right)^{-1}\int_{\Omega_1}\operatorname{div}(v_h)dw_h\int_{\Omega_1}\operatorname{tr}(B^k)dw_h\right)$$
$$+\int_{\partial_c\Omega}\alpha K\left\|\Delta v_{ht}\cdot N^k ds_h\right\| = 0 \tag{5.26}$$

where ρ_{pen} is a large numerical coefficient.

Equation (5.27) is then substituted to Eq. (5.24):

$$\forall 1, \Lambda^1 = -\rho_{\text{pen}}\left(\int_{\Omega_1} dw_h\right)^{-1}\int_{\Omega_1}\operatorname{div}(v_h)dw_h \tag{5.27}$$

Finally, the discrete equations of the problems can be summarized by:

$$\begin{cases}\forall k,\ R^k(X,V) = 0\\ \forall k \in \partial_c\Omega, \begin{cases}(V^k - V_{\text{die}}).\ n^k = 0\\ R^k(X,V)\cdot t^k = 0\end{cases}\end{cases} \tag{5.28}$$

where X and V are the arrays of the nodal coordinates and velocities of the finite element mesh,

At each time step of the process, these non linear equations are solved by a Newton-Raphson algorithm:

$(0)\quad V^{(0)} = 0$ or $V^{(0)}$ is estimated from the previous time step

$(i) \Rightarrow (i+1)\quad V^{(i)}$ is known

$V^{(i+1)} = V^{(i)} + \Delta V^{(i)}$ which is the solution of:

$$R(X, V^{(i+1)}) \approx R(X, V^{(i)}) + \partial R/\partial V(X, V^{(i)})\Delta V^{(i)} = 0 \tag{5.29}$$

$$\text{so: } \Delta V^{(i)} = -\left(\frac{\partial R}{\partial V}(X, V^{(i)})\right)^{-1} R(X, V^{(i)}) \tag{5.30}$$

Contact Algorithm and Time Integration

In order to satisfy the unilateral contact condition, the following algorithm is used. Initially, all the nodes which are geometrically in contact with the dies are treated as contact nodes of $\partial_c\Omega$ [Eq. (5.19)] and the velocity field is so computed. If the calculated normal stress value of any node of $\partial_c\Omega$ is not negative, then the node is released from the contact surface $\partial_c\Omega$, and a new velocity field is computed. This is repeated until all the nodes of $\partial_c\Omega$ exhibit a negative normal stress value.

At time $t + \Delta t$, the domain Ω_t is updated according to:

$$\forall k, \quad X^k_{t+\Delta t} = X^k_t + V^k_t\Delta t \tag{5.31}$$

A node which penetrates the tool during the time interval Δt has to be projected onto the tool surface at time $t + \Delta t$ as shown in Fig. 5.18. This is also the case for a node which belongs to the contact surface at time t and which is then supposed to remain on the tool surface during Δt.

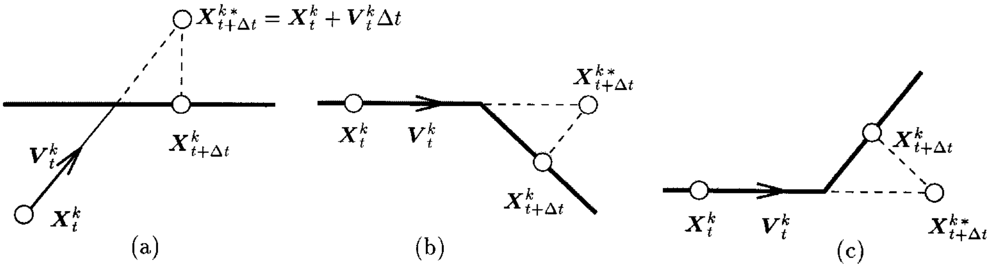


FIGURE 5.18 Contact events: node projected back on the tool surface for a new contact (a), for a node sliding on a concave (b) and convex (c) part of the tool.

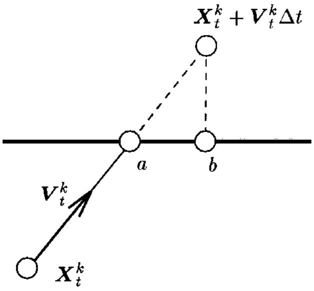


FIGURE 5.19 Time adaptation to the contact events—the time step is reduced in order that the node does not penetrate the tool—it so comes to point (a) rather than being projected onto point (b).

These projections can result in a variation of the volume of material and so numerically disturb the deformation path. In order to minimize them, the time step length can be adjusted in such a way that the new nodes coming into contact do not penetrate the tool (see Fig. 5.19).

Incremental Mass Conservation

At each time step, the computed velocity field is incompressible. However, according to the time integration scheme [(Eq. (5.28)], this field is assumed to remain constant during Δt. This is a first cause for the numerical mass variation. The second cause has already been mentioned: the projections due to the contact treatment. Finally, there is a third cause, the domain variations due to the remeshings, which will be described in the following section.

As it will be discussed in more detail in Section 5.4, the sensitivity analysis requires a very accurate simulation of the process. It is easy to understand that an accurate mass conservation is crucial for the optimal design problem. If we consider a standard forging simulation, at the end of forging, the volume losses can be about 0.5%. If the gap between the desired and the obtained final parts is also about 0.5%, the volume of the flash not being taken into account, it is not possible to conclude whether the design is satisfactory or not. Indeed, the gap can either be due the volume losses or to a bad design or both. So, if we want to control the design optimization, it is important to have a perfect mass conservation.

In order to compensate these volume losses, an incremental mass conservation equation is written, between time $t = t_0$, the beginning of the process, and time t. A numerical density of the material ρ is then introduced to take into account the mass variation:

$$\begin{aligned} \rho_0\, dw_0 &= \rho_{t+\Delta t} dw_{t+\Delta t} \\ \text{but: } dw_{t+\Delta t} &= J^t_{t+\Delta t} dw_t \end{aligned} \tag{5.31}$$

$$\begin{aligned} \text{and: } J_{t+\Delta t} &= 1 + \operatorname{div}(v_t)\Delta t + O(\Delta t^2) \\ \text{So: } \rho_{t+\Delta t} &= 1 \Leftrightarrow (1 + \operatorname{div}(v_t)\Delta t)dw^t = \rho_0\, dw_0 \end{aligned} \tag{5.32}$$

As piecewise functions are used for the pressure interpolation, this equation has to be integrated on each element:

$$\forall 1, \int_{\Omega_1} \mathrm{div}(v_h) dw_h + V_t^1 + \frac{1}{\Delta t}(V_t^1 - \rho^0 V_0^1) = 0 \tag{5.33}$$

where: V_t^1 is the volume of the l element at time t, and V_0^1 is the volume of this element at time $t = t_0$.

This equation is substituted to Eq. (5.25) and introduced in the new penalty formulation:

$$\begin{aligned} &\forall k, \int_{\Omega} 2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-1} \dot{\varepsilon}_h : B^K(dw_h) \\ &+ \rho_{\mathrm{pen}} \sum_1 (V_t^1)^{-1} \left(\int_{\Omega_1} \mathrm{div}(v_h) dw_h - \frac{V_t^1 - \rho_0 V_0^1}{\Delta t} \right) \int_{\Omega_1} \mathrm{tr}(B^k) dw_h \\ &+ \int_{\partial_c \Omega} \alpha K \| \Delta v_{ht} \|^{q-1} \Delta v_{ht} . N^k ds_h = 0 \end{aligned} \tag{5.34}$$

This method makes it possible to reduce the volume losses to less than 0.01% [Fou95b].

Remeshing and Transfer of State Variables

The large deformations of the material drive large deformations of the finite element mesh which get distorted and badly adapted to the problem. So, as soon as one of the following remeshing criteria has been reached, a new mesh has to be automatically generated:

- distortion of a finite element, or distortion of a segment of the boundary,
- detection of a fold (see Fig. 5.20),
- penetration of a part of the mesh inside the tool: as the contact condition is enforced only for the nodes of the mesh, some parts of boundary segments may penetrate the tools while the nodes cannot (see Fig. 5.21); this means that the mesh cannot properly describe the tool/workpiece interface,
- large curvature of the segments of the free boundary.

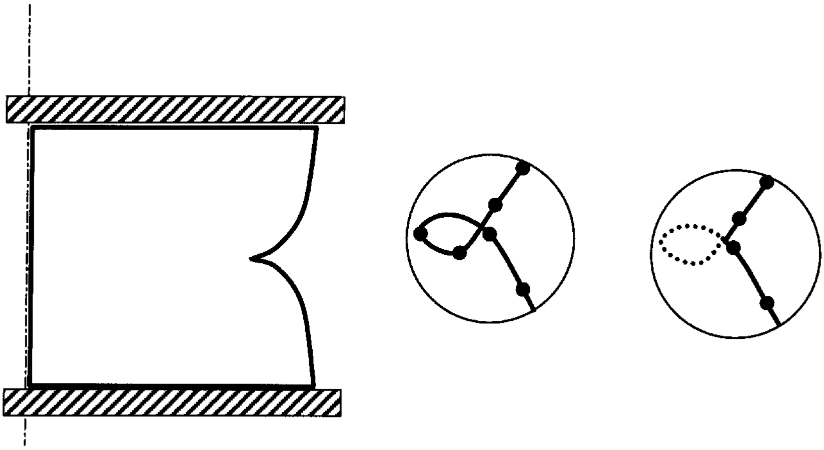


FIGURE 5.20 Folding defect noticed by the remeshing procedure when two boundary segments cross each other—Elimination of the folded area by the remeshing procedure.

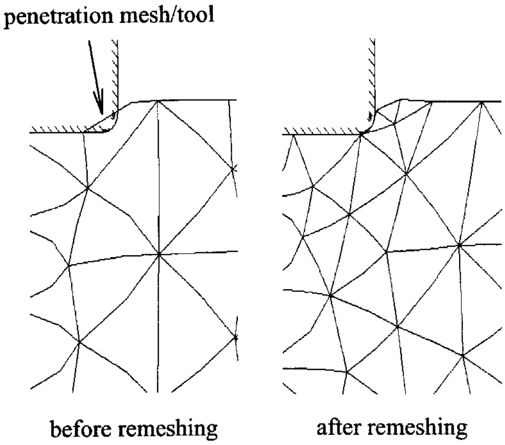


FIGURE 5.21 Penetration of the finite element mesh into the tool—New adapted mesh generated, and corresponding volume losses.

The remeshing procedure is based on the following three steps:

- the extraction of the boundary of the previous mesh,
- its adaptation to the contact with the tool surface and the optimization of the number and location of its nodes,
- the generation of a new triangular mesh based on this boundary.

The optimization step makes it possible to satisfy all the boundary mesh criteria, and more particularly those used to trigger the remeshing. If a fold has been detected, the overlapping part of the domain is removed from the boundary description (see Fig. 5.20). The triangular mesh generation is based on a Delaunay algorithm. A first tessellation is produced from the boundary set of points, then internal nodes are added and tessellated. Finally, the mesh quality is improved using regularization and topological improvement algorithms [18].

The optimization of the position of the boundary nodes results in a difference of volume between the two meshes. Although the algorithm tries to minimize it, when the number of remeshings is large or when the meshes are coarse, the resulting loss of volume may be non negligible. This phenomenon is even increased when a fold has been removed. In order to recover this volume variation, the p_0 numerical density has been introduced in the incremental mass conservation scheme. It is equal to the ratio between the actual volume of the workpiece and the current volume of the workpiece just after remeshing [27].

Finally, once the new mesh has been created, the state variables have to be transferred onto it. With the present formulation, the problem does not have state variables but the mesh coordinates. However, for the optimization problem, the $\frac{dX}{dp}$ derivatives have to be transferred. As already mentioned, the optimization problem is sensitive to the accuracy of the calculations, so the accurate Liszka-Orkisz finite difference method is used [32, 54]. For any node P of the new mesh, we look for the nearest node M of the old mesh. We also select a set of neighboring nodes V of the old mesh, which surround M. For instance, the V and M nodes may belong to the same element. Then, a second order Taylor series expansion is written for any node V and any f variable to be transferred:

$$f(V) = \tilde{f}(V) + O(\| \Delta x_v \|^3) \tag{5.35}$$

$$\text{with:} \quad \tilde{f}(V) = f(P) + \frac{df}{dx}(P)\Delta x_v + (1/2)\Delta x_V^t \frac{d^2 f}{dx^2}(P)\Delta x_v \tag{5.36}$$

$$\text{and:} \quad \Delta x_V = x_v - x_p \tag{5.37}$$

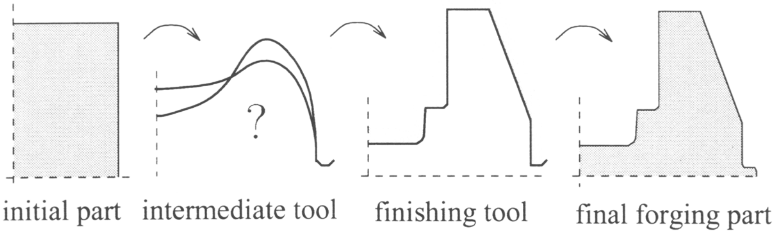


FIGURE 5.22 A two-step forging sequence—The initial billet is a simple cylinder, the shapes of the final dies are derived from the shape of the desired part—The shapes of the preforming dies are unknown.

The $f(P)$ value minimizes the following least square functional:

$$\Phi_{\text{Ork}}\left(f(P),\frac{df}{dx}(P),\frac{d^2f}{dx^2}(P)\right)=\sum_V\frac{\|f(V)-\tilde{f}(V)\|^2}{\|\Delta x_v\|^6} \tag{5.38}$$

which just requires the resolution of a local linear system.

Forging Optimization Problem

In many cases, a part cannot be forged in a single operation, and the deformation path must be studied in order to avoid the material folds or other defects. Intermediate forging steps are also introduced to reduce the forging force, or to allow a deformation which cannot be obtained in a single stroke. Specific criteria may also motivate complex deformation routes, such as those already mentioned in Section 5.2. Very often, the shape of the initial billet is simple, most of the time a cylinder. The shape of the finishing die is known as well, as it almost corresponds to the desired part. So, most of the design work concerns the shape of the preforming tools, and the process parameters (see Fig. 5.22).

Objective Function, Regularization Functions, and Constraints

Prescribed Shape

When designing a sequence of forging tools, the primary goal is to completely fill the finishing tools at the last stage of the process. This criterion is important since when it is not respected, the part is rejected. This can either be due to a bad distribution of the material flow or to a piping phenomenon (see Fig. 5.24). A mathematical measure of this criterion can be obtained at the end of the simulation by comparing the outline of the desired part Ω_{des} and that of the actually forged part Ω_{end} (see Fig. 5.23). The volume of the zones where the two outlines do not coincide is a measure of this criterion. However, a zone where the two outlines are distinct may be either an unfilled area or a material surplus—typically a flash. As long as the volume of the flash does not have to be optimized, the corresponding overlap defect is ignored.

$$\Phi_{\text{fil}}=\int_{\Omega_{\text{end}}\cup\Omega_{\text{des}}\setminus\Omega_{\text{end}}\cap\Omega_{\text{des}}}1_{\text{desired}}\,dw \tag{5.39}$$

where

Ω_{end} represents the part at the end of the process,

Ω_{des} represents the desired part which should be obtained,

1_{desired} is a characteristic function which can be introduced when the shape of the flash is not considered: it is equal to 1 when the point belong to Ω_{des} and not to Ω_{end}, and is equal to 0 elsewhere.

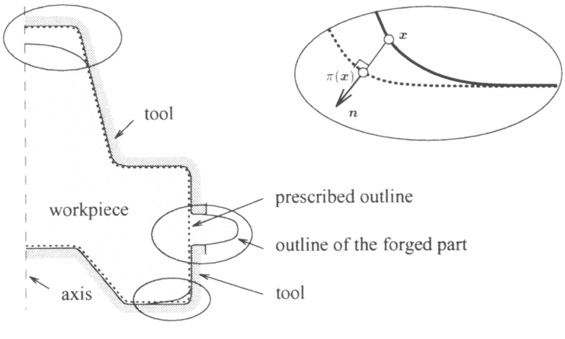


FIGURE 5.23 Filling defect—The volume of the flash may be not considered.

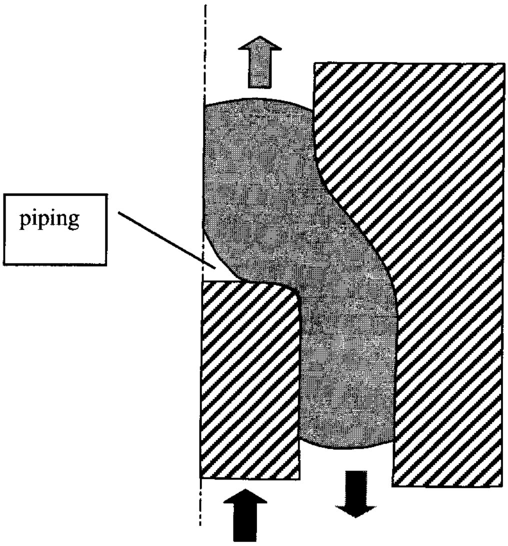


FIGURE 5.24 Piping defect due to a bad study of the material flow.

After the finite element discretization, the overlap defects are closed loops in which nodes belong either to the desired outline or to the forged part boundary. Then, the overlapping volume can be computed with:

$$\Phi_{\text{fil},\,h} = \frac{\pi}{3}\sum_{k\in\partial\Omega_{\text{end}}\backslash\partial\Omega_{\text{def}}} 1_{\text{desired}}(Z_{k+1}-Z_k)(R_{k+1}^2+R_k^2+R_kR_{k+1}) \tag{5.40}$$

where (R, Z) are the cylindrical coordinates of the node k and $\partial\Omega_{\text{def}}$ denotes the boundary of the overlap defects.

Forming Energy

Often, the optimization problem is not well posed, as several solutions may minimize the objective function. So, it is better to introduce a regularization function in order to reduce the number of such solutions. The total energy of the forming process is such a function. Moreover, it covers a real concern of the designer.

$$\Phi_{\text{ene}} = \int_{t=t_0}^{t=t_{\text{end}}}\int_{\Omega_t} K(\sqrt{3}\dot{\bar{\varepsilon}})^{m+1}(dw)\int_{\partial_c\Omega_t}\alpha K\|\Delta v_t\|^{q+1}ds\,dt \tag{5.41}$$

The maximum forging force is a more interesting objective function (or constraint). Actually, the press capacity is limited and the reduction of the required force can make it possible to forge the part using a lower capacity press. This can also help to decide whether the number of forging operations can be reduced or not.

$$\Phi_{\max} = \underset{t}{\mathrm{MAX}}\left\{ \frac{\left(\int_{\Omega_t} K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m+1} dw + \int_{\partial_c \Omega_t} \alpha K \| \Delta v_t \|^{q+1} ds \right)}{V_{\mathrm{die},\,t}} \right\} \tag{5.42}$$

where $V_{\mathrm{die},t}$ is the generalized tool velocity at time t.

Tool Wear

For mass production, the tool wear can be a critical criterion, because the tool life has a direct influence on the production costs. For abrasive wear, the following objective function can be suggested [Fel89]:

$$\Phi_{\mathrm{wear}} = \int_{\partial\Omega_{\mathrm{die}}} \int_{t=t_0}^{t=t_{\mathrm{end}}} K_{\mathrm{wear}} \Delta v_t \sigma_n dt ds \tag{5.43}$$

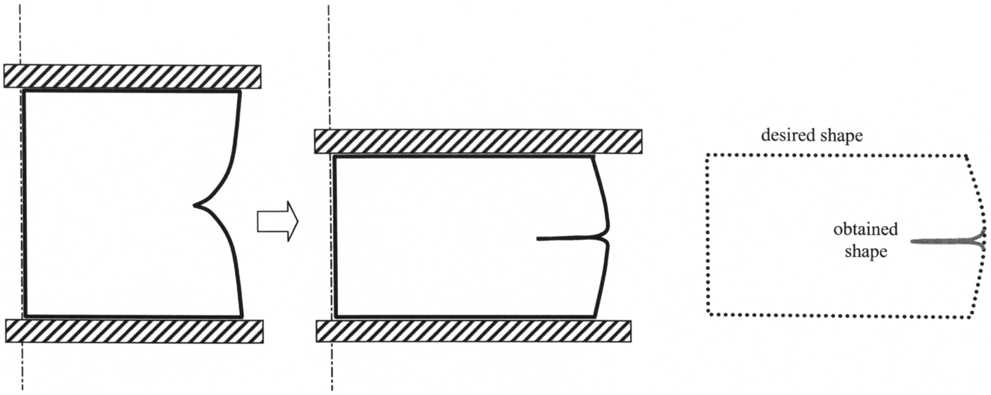
where K_{wear} is a wear coefficient, and σ_n is the normal stress.

Folding Defects

During the first stages of the design process, avoiding folding defects are probably the main concern of the designer. The corresponding objective function (or design constraint) can be derived from the distance of the folded boundary points from the desired shape (see Fig. 5.25):

$$\Phi^{I}_{\mathrm{fold}} = \int_{\partial\Omega} \| x - \pi_{\mathrm{des}}(x) \| ds \tag{5.44}$$

where $\pi_{\mathrm{des}}(x)$ is the projection of any point x on $\partial\Omega_{\mathrm{des}}$.



Intermediate shape

Final shape

Desired versus obtained shapes

FIGURE 5.25 At the end of the simulation, as a result of a folding defect, a part of the boundary is not in contact with the desired shape.

It can also be noticed that the material is so severely sheared in the folding zone that there is an unusual increase of the strain rates and the strains there. Therefore, another objective function can be derived from the maximum values of the strain rates in the folded zones which have been eliminated by the mesh generator:

$$\Phi^2_{\text{fold}} = \sum_{\text{folds}} \underset{x \in \text{fold}}{\text{MAX}} \dot{\bar{\varepsilon}}(x) \tag{5.45}$$

or from the strain values which have exceeded a limit value $\bar{\varepsilon}_{\text{lim}}$ onto the boundary:

$$\Phi^3_{\text{fold}} = \int_{\partial\Omega} [\bar{\varepsilon} - \bar{\varepsilon}_{\text{lim}}]^+ ds \tag{5.46}$$

where $[x]^+$ represents the positive part of the variable x.

Metallurgical Aspects

For security parts, a great attention has to be paid to the metallurgical aspects during the design process. For instance, it is important to obtain an homogeneous and small grain size distribution in order to produce a part with a high resistance to fatigue. Complex grain size evolution laws can be introduced in the simulation software. However, in a simple way, a homogeneous distribution of the strains can be a satisfactory approximated criterion. So the following objective functions can be proposed:

$$\Phi^1_{\text{met}} = \bar{\varepsilon}_{\max} - \bar{\varepsilon}_{\min} \tag{5.47}$$

or

$$\Phi^2_{\text{met}} = \int_{\partial\Omega_{\text{end}}} (\text{grad}(\bar{\varepsilon}))^2 dw \tag{5.48}$$

Other functions can be defined in order to address other metallurgical aspects such as solid recristallized fraction and fibering.

Constraints

Depending on the design objectives and the qualities of the initial design, some functions can be either considered as objectives or as constraints. There are also some specific constraints for the parameter values, such as geometrical constraints on the shape of the tools in order to ensure that the tool can be machined and that it makes it possible to extract the workpiece. Links between shape parameters, for instance in order to keep constant the volume between the dies in close-die forging, can also be mentioned. There are also the process constraints like the maximum admissible press velocity, the maximum workpiece and dies temperature, etc.

Design Parameters

When the initial shape of the billet is also a design parameter, it can be either described by its height/radius ratio, as often it is a cylinder, or in the same way as the dies.

Arcs and Lines

In the industrial practice, the tool shapes are defined using various CAD systems, so the best user-friendly approach would be to derive a discretization of these shapes from the CAD systems. For two-dimensional problems, the dies are often designed using a succession of segments and pieces of arcs, the arcs ensuring the continuity of the normal of the surface. In this case, the p1 shape parameters simply are the end positions of the segments, and the radius of the pieces of arcs (see Fig. 5.26).

Splines and Bsplines

A description with spline functions [1] has several advantages: it is well adapted to CAD systems, it automatically ensures the required continuity of the die surface, it reduces the number of optimization parameters, and it makes it easier to introduce the physical constraints of the problem.

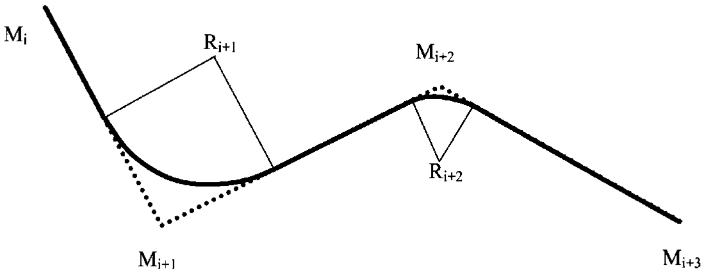


FIGURE 5.26 Tool discretization using arcs and lines.

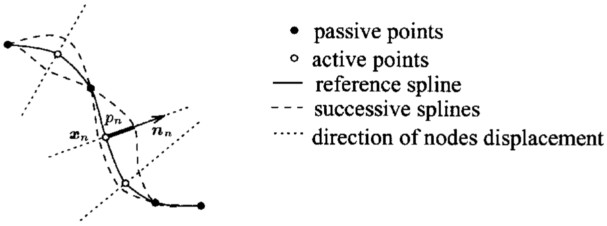


FIGURE 5.27 Spline discretization of the tool shapes.

The shape co-ordinates functions (x_1^s, x_2^s) are defined according to:

$$\forall n,\quad \xi^n < \xi < \xi^{n+1}:\quad \begin{cases} x_1^s(\xi) = \sum_{k=1}^{2d} C_{x_1}^{k,n} u^k(\xi - \xi^n) \\ x_2^s(\xi) = \sum_{k=1}^{2d} C_{x_2}^{k,n} u^k(\xi - \xi^n) \end{cases} \tag{5.49}$$

where

$$u^k(\xi) = \frac{1}{(k-1)!}\xi^{k-1} \tag{5.50}$$

2d−1 is the degree of the spline function which is generally taken as three, $C_{x_1}^{k,n}$ and $C_{x_2}^{k,n}$ are the spline coefficients which are defined in order that:

$$\begin{cases} x_1^s(\xi^n) = x_1^s \quad \text{and} \quad x_2^s(\xi^n) = x_2^n \\ \left(\dfrac{d^p x}{d\xi^p}\right)_{p=1,2} \quad \text{is also continuous} \end{cases} \tag{5.51}$$

$(x_1^n, x_2^n)_n$ are the characteristic points which define the spline.

The optimization parameters are the displacements of some of these characteristic points along selected directions—for example, the directions normal to the initial spline, as shown in Fig. 5.27.

A shortcoming of cubic splines is that the displacement of a single point at the one hand of the curve modifies the curve everywhere, so cubic Bspline functions can also be used, as the displacement of a

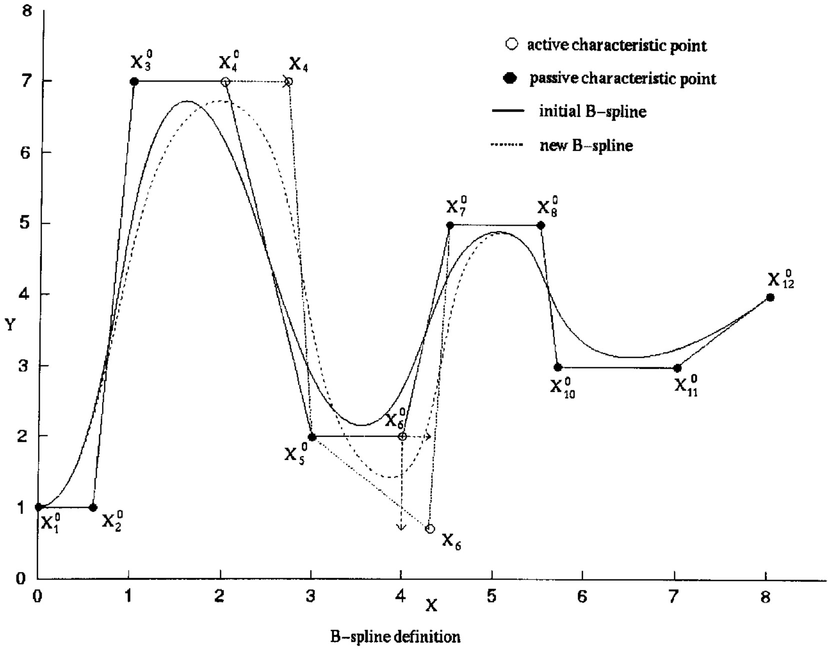


FIGURE 5.28 Interpolation of the tool shape using B spline functions—active and passive control points—modified shape after displacement of an active point.

master point has only local effects. The cubic Bsplines approximation is defined as follows:

$$\forall s \in [s_{\min}, s_{\max}], \qquad \boldsymbol{P}_B(s) = \sum_{i=1}^{N} M_{i,4}(s)\boldsymbol{x}_i \tag{5.52}$$

where

s is the curvilinear abciss of point $\boldsymbol{P}_B(s)$,
$\boldsymbol{x}_i$ are the coordinates of the characteristic points of the Bspline,
N is the number of characteristic points,
$M_{i,4}$ are the cubic Bspline basis functions, which are recursively defined [Cox71]:

$$\text{if } s \in [s_i, s_{i+1}] \text{ then } M_{i,1}(s) = 1, \text{ else } M_{i,1}(s) = 0$$

$$\text{and then, } \forall k, M_{i,k}(s) = \frac{s - s_i}{S_{i+k-1} - S_i} M_{i,k-1}(s) + \frac{s_{i+k} - s_i}{S_{i+k} - S_{i+2}} M_{i+1,k-1(s)} \tag{5.53}$$

Here also, the shape parameters are the displacement of some selected characteristic points of the curve, which can be active either in a single arbitrary space direction or in both space directions.

Process Parameters

As already mentioned, several process parameters p_2 can be optimized during the process design, such as:

- the kinematics of the different tools, this is to say the press velocity and the forging length, and possibly the relative motion of dies,
- the initial temperature of the workpiece and the different dies, and possibly the transfer time between operations and the characteristics of the possible intermediate heat treatments.

Optimization Problem

So, as mentioned in Section 5.2, the optimization problem is written:

$$\text{find } p \text{ such that: } \Phi(p) = \underset{p',\begin{cases} C_1(p') = 0 \\ C_2(p') \leq 0 \end{cases}}{\text{MIN}} \Phi(p') \tag{5.54}$$

Φ can be one of the functions previously described. However, it is important to check that the problem is well posed, which might not be the case with all the proposed functions. If this property is not satisfied, the optimization method, whichever it is, will not work. Several techniques can be used to obtain a well posed problem, either by introducing more constraints, or by adding a regularization function:

- energy method: among the possible solutions, the selected solution minimizes the forming energy; as already mentioned, often this is also an actual objective of the design

$$\Phi_{\text{reg}}(p) = (1 - \lambda)\Phi(p) + \lambda\Phi_{\text{ene}}(p) \tag{5.55}$$

where λ is a regularization parameter

- Tikhonov method [62]: the selected solution is the nearest solution from the initial set of parameters

$$\Phi_{\text{reg}}(p) = (1 - \lambda)\Phi(p) + \lambda\sum_{i=1}^{N_{\text{par}}}(p^i - p_0^i)^2 \tag{5.56}$$

where

N_{par} is the number of the parameter values, $(p^i)_i$,

(p_0^i) are the initial values of $(p^i)_i$.

Often, the design objectives are not unique and several of them have to be considered in the optimization procedure. A first approach is to decompose the problem into several sub-problems. This is possible if the objectives are not too strongly coupled, or if it is possible to define a clear hierarchy between them. The method can be first applied to a small number of important objectives, and then, starting again from the optimized design, the other objective functions are progressively introduced. During the second step, the previous functions can be considered as constraints, in order to prevent too large variations of their values. For instance, it is possible to first optimize the shape of the preforming dies in order to obtain the right final part without any defects, and then to optimize the process parameters such as the initial temperature of the billet, the temperature of the dies, and the forging velocity, in order to optimize the thermo dependent metallurgical properties of the part, for instance by minimizing the grain size. However, when the functions are strongly coupled or when it is expensive to carry out to many optimizations, a multi-objective function can be defined:

$$\Phi_{\text{tot}}(p) = \sum_{i=1}^{N_{\text{obj}}}\lambda_i\Phi_i(p) \tag{5.57}$$

where

N_{obj} is the number of different objective functions, Φ_i,

λ_i are the weighting coefficients, taking into account both the physical dimensions of Φ_i, its interval of variations, and its priority with respect to the other functions.

Obviously, the choice of the λ_i coefficients is delicate and will influence the final solution.

5.4 Sensitivity Analysis–Optimization Method

The convergence rate of the optimization methods can be dramatically increased by computing the first order gradients of the objective and constraint functions, Φ and C, as it allows us to use the efficient quasi-Newton algorithms such as the well-known DFP or BFGS algorithms [20]. Actually, in the application examples of Section 5.5, we will show that satisfactory solutions can be obtained after just a few iterations.

We shall describe here the main two approaches for computing these gradients: the adjoint state method and the direct method. Both are developed at the discrete level, rather than using a variational formulation. A special focus is put on the direct method which has been used for the applications of Section 5.

Adjoint and Direct Methods

Adjoint methods have been widely used in structural mechanics. They have also been proposed by some authors for steady-state forming problems [13]. Direct and adjoint methods exhibit the same accuracy in the calculations of the sensitivities. In fact, they just differ by their computational efficiency which depends on the ratio between the number of parameters and the number of objective (and constraints) functions, and the method used to solve linear systems.

Steady-State Formulation

In a first step, the time evolution is neglected. The objective function can be described as an explicit function of the velocity field V, mesh coordinates X, and parameters p:

$$\Phi(p) = (\Phi V(p), X(p), p) \tag{5.58}$$

So:

$$\frac{d\Phi}{dp} = \frac{\partial\Phi}{\partial V}\frac{dV}{dp} + \frac{\partial\Phi}{\partial X}\frac{dX}{dp} + \frac{\partial\Phi}{\partial p} \tag{5.59}$$

For a given optimization problem, dX/dp is known, or at least can be calculated. $\partial\Phi/\partial V$, $\partial\Phi/\partial X$, and $\partial\Phi/\partial p$ can be straightforwardly computed by differentiating the expression of Φ. So, the main problem is to compute dV/dp. In order to do so, the simplified form of the discretized equilibrium equation is used and differentiated:

$$R(X,V) = R(V(p), X(p), p) = 0, \quad \forall p \tag{5.60}$$

So:

$$\frac{\partial R}{\partial V}\frac{dV}{dp} + \frac{\partial R}{\partial X}\frac{dX}{dp} + \frac{\partial R}{\partial p} = 0 \tag{5.61}$$

and then:

$$\frac{dV}{dp} = -\left(\frac{\partial R}{\partial V}\right)^{-1}\left(\frac{\partial R}{\partial X}\frac{dX}{dp} + \frac{\partial R}{\partial p}\right) \tag{5.62}$$

dV/dp is solution of a linear system with the same matrix $\partial R/\partial V$ as the stiffness matrix obtained at the last iteration of the Newton-Raphson method for the forging problem [(Eq. (5.27)]. Actually, there are N_{par} linear systems, as many as parameters. If a direct resolution method has been used for the direct forging problem, then the matrix $\partial R/\partial V$ has already been triangularized, if a direct method is utilized, so the computational effort for the resolution of these systems is rather negligible and does not depend much on the number of parameters.

On the other hand, with the adjoint state method, another linear system is solved. Indeed, this method is based on the computation of a Lagrange parameter λ such as:

$$\frac{\partial R}{\partial V}\lambda = -\frac{\partial\Phi}{\partial p} \tag{5.63}$$

because λ makes it possible to eliminate dV/dp from the following Eq. (5.67) but first, the Lagrangian function is introduced as:

$$\Lambda(p, \lambda) = \Phi(p) + \lambda R(V(p), X(p), p) \tag{5.64}$$

and differentiated:

$$\frac{d\Lambda}{dp} = \frac{d\Phi}{dp} + \lambda \frac{dR}{dp} \tag{5.65}$$

but, as in Eq. (5.61),

$$\frac{dR}{dp} = 0, \quad \text{so} \quad \frac{d\Lambda}{dp} = \frac{d\Phi}{dp} \tag{5.66}$$

using the chain rule:

$$\frac{d\Lambda}{dp} = \frac{\partial \Phi}{\partial p} + \left(\frac{\partial \Phi}{\partial X} + \lambda \frac{\partial R}{\partial X}\right)\frac{dX}{dp} + \left(\frac{\partial \Phi}{\partial V} + \lambda \frac{\partial R}{\partial V}\right)\frac{dV}{dp} \tag{5.67}$$

so according to the definition of λ (5.63):

$$\frac{d\Phi}{dp} = \frac{\partial \Phi}{\partial p} + \left(\frac{\partial \Phi}{\partial X} + \lambda \frac{\partial R}{\partial X}\right)\frac{dX}{dp} \tag{5.68}$$

where all the values are known, but the Lagrange parameter λ.

Therefore, with the adjoint method, the same system matrix is obtained but there are as many systems as objective and constraint functions. This method shows more efficient when the number of parameters exceed the number of functions to be differentiated, which is often the case. The computational time difference with the direct method can be significant if an iterative solver is used to solve the linear systems, because then the computational cost is proportional to the number of systems solved.

Non Steady-State Extension of the Adjoint Method

Generally, the adjoint method is applied to steady state problems as in [55]. However, it can be extended to non steady state problems, as suggested in [63], although the method becomes much more complex. In order to make this presentation more simple, we assume that the objective function depends only on X_{tend}, the shape of the domain at the end of the simulation. For instance, Φ_{fil} can be considered. So:

$$\Phi(p) = \Phi(X_{\text{tend}}(p)) \tag{5.69}$$

According to the time integration equation:

$$X_{\text{tend}} = X_{t0} + \sum_{i=0}^{N_{\text{ine}}-1} V_i \Delta t_i \tag{5.70}$$

where N_{inc} is the total number of time increments for the simulation, and V_i and Δt_i are the velocity and the time increment of increment i.

It is then necessary to introduce as many Lagrange parameters as time increments:

$$\Lambda(p, \lambda) = \Phi(p) + \sum_{i=0}^{N_{\text{ine}}} \lambda_i R_i(V(p), X(p), p) \tag{5.71}$$

after differentiation:

$$\frac{d\Lambda}{dp} = \frac{\partial \Phi}{\partial p} + \frac{\partial \Phi}{\partial X_{\text{tend}}}\frac{dX_{\text{tend}}}{dp} + \sum_{i=0}^{N_{\text{ine}}} \lambda_i \left(\frac{\partial R_i}{\partial V_i}\frac{dV_i}{dp} + \frac{\partial R_i}{\partial X_i}\frac{dX}{dp} + \frac{\partial R_i}{\partial p}\right) \tag{5.72}$$

and
$$\frac{dX_{\text{tend}}}{dp} = \frac{dX^{t0}}{dp} + \sum_{i=0}^{N_{\text{inc}}-1} \frac{dV_i}{dp}\Delta t_i \tag{5.73}$$

so
$$\begin{aligned}\frac{d\Phi}{dp} &= \sum_{i=0}^{N_{\text{inc}}} \lambda_i \frac{\partial R_i}{\partial p} + \left(\frac{\partial \Phi}{\partial X_{\text{tend}}} + \sum_{i=0}^{N_{\text{inc}}} \lambda_i \frac{\partial R_i}{\partial p}\right)\frac{dX_{\text{tend}}}{dp} \\ &+ \sum_{i=0}^{N_{\text{Ninc}}-1}\left(\left[\frac{\partial \Phi}{\partial X_{\text{tend}}} + \sum_{j=i+1}^{N_{\text{inc}}} \lambda_j \frac{\partial R_j}{\partial X_j}\right]\Delta t_i + \lambda_i \frac{\partial R_i}{\partial V_i}\right)\frac{dV_i}{dp} + \lambda_i \frac{\partial R_{\text{Ninc}}}{\partial V_{\text{Ninc}}}\frac{dV_{\text{Ninc}}}{dp}\end{aligned} \tag{5.74}$$

The Lagrange parameters are chosen in order to remove the dV_i/dp terms from the previous equation, so they are recursively defined by:

$$\lambda_{N\text{inc}} = 0 \tag{5.75}$$

$$\lambda_{\text{Ninc}-1}\frac{\partial R_{\text{Ninc}-1}}{\partial V_{\text{Ninc}-1}} = -\Delta t_{\text{Ninc}-1}\left(\frac{\partial \Phi}{\partial X_{\text{tend}}} + \lambda_{\text{Ninc}}\frac{\partial R_{\text{Ninc}}}{\partial X_{\text{Ninc}}}\right) \tag{5.76}$$

$$\lambda_{\text{Ninc}-2}\frac{\partial R_{\text{Ninc}-2}}{\partial V_{\text{Ninc}-2}} = -\Delta t_{\text{Ninc}-2}\left(\frac{\partial \Phi}{\partial X_{\text{tend}}} + \lambda_{\text{Ninc}}\frac{\partial R_{\text{Ninc}}}{\partial X_{\text{Ninc}}} + \lambda_{\text{Ninc}-1}\frac{\partial R_{\text{Ninc}-1}}{\partial X_{\text{Ninc}-1}}\right) \tag{5.77}$$

$$\lambda_0\frac{\partial R^0}{\partial V^0} = -\Delta t^0\left(\frac{\partial \Phi}{\partial X_{\text{tend}}} + \sum_{j=1}^{\text{N}_{\text{inc}}} \lambda^j \frac{\partial R^j}{\partial X^j}\right) \tag{5.78}$$

And then
$$\frac{d\Phi}{dp} = \sum_{i=0}^{\text{N}_{\text{inc}}} \lambda_i \frac{\partial R_i}{\partial p} + \left(\frac{\partial \Phi}{\partial X_{\text{tend}}} + \sum_{i=0}^{\text{N}_{\text{inc}}} \lambda_i \frac{\partial R_i}{\partial X_i}\right)\frac{dX_{\text{tend}}}{dp} \tag{5.79}$$

At each time increment, the derivative calculations are similar to those of the steady process. However, they cannot be carried out at the same time as the resolution of the direct forging problem, because the calculations have to be started from the last time increment and recursively until the first increment. This means that the results of the direct calculations have to be stored. If there is enough memory space available, the matrix (and possibly its triangularized form) can be kept. This can be quite memory space expensive, and often these information are only stored for selected time increments. Otherwise, the matrix has to be calculated again from the stored state variables (at least the X and V values), which, have to be stored for all the time increments anyway.

According to its complexity and storage problems, this method has not been used for non steady-state forming processes, and the direct method is much more attractive. However, it is an elegant way to reduce the derivative computational time when the number of parameters largely exceeds the number of functions and when an iterative solver is used.

Discrete Direct Method

This section is focused on a more precise description of the discrete direct differentiation method, which seems to be the more effective, having produced most of the present achievements in this area [15, 29, 70]. As a matter of fact, some of the previous equations are presented again with more details.

Derivatives Calculation

In order to simplify the derivative calculations and focus on their basic ideas, this presentation is carried out in the same frame of hypothesis as in Section 5.3.

According to the velocity formulation which has been used to solve the forging problem equations, the different variables of the objective and constraint functions, such as Δv_t, $\dot{\bar{\varepsilon}}$, $\bar{\varepsilon}$, σ, … can be expressed as functions of the velocities and coordinates. Consequently, Φ and C can be considered as functions explicit of the optimization parameters, p, x, and v. In the following, only the objective function will be considered while the same method applies to the constraint function.

$$\Phi(p) = \Phi(v(p), x(p), p) \tag{5.80}$$

For the discrete method, the discretized form of Φ is considered:

$$\Phi_h(p) = \sum_{i=0}^{\mathrm{N_{inc}}} \Phi_i(V_i(p), X_i(p), p) \tag{5.81}$$

where Φ_i is the contribution to the Φ value which is computed at increment i, for instance:

$$\text{for } \Phi_{\text{fil}},\ \text{ if } i < N_{\text{inc}}\text{, then } \Phi_i = 0 \quad \text{while} \quad \Phi_{\mathrm{Ninc}} = \int_{\Omega_{\text{end}} \cup \Omega_{\text{des}} \backslash \Omega_{\text{end}} \cap \Omega_{\text{des}}} 1_{\text{desired}} dw \tag{5.82}$$

$$\text{for} \quad \Phi_{\text{ene}}, \forall i, \Phi_i = \left(\int_{\Omega_t} K(\sqrt{3}\dot{\bar{\varepsilon}})^{m+1} dw + \int_{\partial_c \Omega_i} \alpha K \| \Delta v_t \|^{q+1} ds \right) \Delta t_i \tag{5.83}$$

The chain rule differentiation gives:

$$\frac{d\Phi_h}{dp} = \sum_{i=0}^{\mathrm{N_{inc}}} \frac{\partial \Phi_i}{\partial V_i} \frac{dV_i}{dp} + \frac{\partial \Phi_i}{\partial X_i} \frac{dX_i}{dp} + \frac{\partial \Phi_i}{\partial p} \tag{5.84}$$

The partial derivatives $\partial \Phi_i / \partial V_i$, $\partial \Phi_i / \partial X_i$ and $\partial \Phi_i / \partial P$, are straightforwardly calculated from the differentiation of the discrete equations which define the objective function. On the other hand, dV_i/dp is calculated from the differentiation of the discrete equations of the forging problem, while dX_i/dp is obtained by the time integration of dV_i/dp. As the direct forging problem is solved incrementally, the sensitivity analysis is carried out in the same way.

Incremental Calculation of the Basic Derivatives

In fact, at time $t = 0$, dX_0/dp is known. It is not equal to zero only if the initial shape of the workpiece is also to be optimized. In this specific case, dX_0/dp is obtained by the differentiation of the functions which parameterize the shape and provide $X_0(p)$. Possibly the meshing itself could also have to be differentiated, but it is a more complex issue which has been addressed in structural mechanics for instance, and which is not relevant for most of the present studied problems.

For any time increment i, dX_i/dp is assumed to be known, so dV_i/dp can be calculated as it will be described. Then, dX_{i+1}/dp is provided by differentiation of the time integration scheme. For the one step Euler scheme, we have:

$$X_{i+1} = X_i + V_i \Delta t_i \tag{5.85}$$

$$\text{so:} \quad \frac{dX_{i+1}}{dp} = \frac{dX_i}{dp} + \frac{dV_i}{dp} \Delta t_i \tag{5.86}$$

However, this equation does not apply to the boundary nodes which have been projected back on the tool surface at time $t + \Delta t$. In this case, a rather more complex equation has to be derived from the projection condition. We note:

$$P = X_{i+1}^{0} = X_i + V_i \Delta t_i \tag{5.87}$$

$$N = X_{i+1} \tag{5.88}$$

where N is the point of the parameterized tool which coincides with X_{i+1}, so there is a curvilinear abciss ξ such that $N(\xi)$; moreover, $N(\xi)$ satisfies:

$$PN \cdot t = 0 \tag{5.89}$$

$$\frac{dN}{dp} = \frac{\partial N}{\partial p} + \frac{\partial N}{\partial \xi}\frac{d\xi}{dp} \tag{5.90}$$

where most of the terms except $d\xi/dp$ can easily be calculated by the differentiation of the parameterization functions of the tool shape. Moreover, if we note:

$$u = \frac{\partial N}{\partial \xi} \tag{5.91}$$

then

$$u = \|u\| t \tag{5.92}$$

Equation (5.89) is multiplied by $||u||$ and then differentiated, which gives:

$$\left(\frac{dN}{dp} - \frac{dP}{dp}\right) \cdot u + PN \cdot \frac{du}{dp} = 0 \tag{5.93}$$

According to Eq. (5.90), where u is substituted to $\partial N/\partial \xi$, Eq. (5.93) provides $d\xi/dp$:

$$\frac{d\xi}{dp} = \frac{1}{\|u\|^2}\left[\left(\frac{dP}{dp} - \frac{\partial N}{\partial p}\right) \cdot u + NP \cdot \frac{du}{dp}\right] \tag{5.94}$$

$$\text{As} \quad NP \bullet \left(\frac{du}{dp}\right) = \|u\|\, NP \cdot \frac{dt}{dp} \tag{5.95}$$

Finally

$$\frac{dX_{i+1}}{dp} = \frac{dN}{dp} = \left(\frac{\partial N}{\partial p} \cdot n\right) n + \left[\frac{dP}{dp} \cdot t + NP \cdot \frac{dt}{dp}\right] t \tag{5.96}$$

Calculation of dV/dp

In order to compute dV_i/dp, at any time increment i, the discrete equations of the problems which have been summarized by Eq. (5.24) are recalled here. For simplicity reasons, the time indexes have been removed:

$$\begin{cases} \forall k \notin \partial_c \Omega, & R^k(V, X, p) = 0 & \text{(5.97a)} \\ \forall k \in \partial_c \Omega, & \begin{cases} (V^k - V_{\text{die}}) \cdot n^k = 0 \\ R^k(V, X, p) \cdot t^k = 0 \end{cases} & \text{(5.97b)} \end{cases}$$

So, a local rotation is imposed to the coordinates of the contact nodes. It can be formulated as follows:

$$\tilde{V} = QV \tag{5.98}$$

$$\tilde{R} = QR \tag{5.99}$$

where Q is a rotation matrix such as:

$$\begin{cases} \forall k \notin \partial_c\Omega, & QV^k = V^k \quad \text{(identity matrix)} \\ \forall k \in \partial_c\Omega, & QV^k = Q^k V^k \quad \text{(local rotation matrix)} \end{cases} \tag{5.100}$$

where we make a common abuse by using the same notation V^k for the restriction of V at node k and for the velocity of node k

$$Q^k = \begin{bmatrix} t_1^k & t_2^k \\ n_1^k & n_2^k \end{bmatrix} \tag{5.101}$$

So Eq. (5.97b) can be summarized by:

$$\begin{cases} \tilde{R}(\tilde{V}, X, p) = 0 \\ \forall k \in \partial_c\Omega, (V^k - V_{\text{die}}) \cdot n^k = 0 \end{cases} \tag{5.102}$$

After differentiation with respect to p:

$$\begin{cases} \dfrac{dQ}{dp}R + Q\left(\dfrac{\partial R}{\partial V}\dfrac{dV}{dp} + \dfrac{\partial R}{\partial X}\dfrac{dX}{dp} + \dfrac{\partial R}{\partial p}\right) = 0 \\ \forall k \in \partial_c\Omega, \quad \dfrac{dV}{dp} \cdot n^k = (V^k - V_{\text{die}}) \cdot \dfrac{dn^k}{dp} \end{cases} \tag{5.103}$$

but

$$Q\,\frac{\partial R}{\partial V}\,\frac{dV}{dp} = Q\frac{\partial R}{\partial V}\,Q^t Q\frac{dV}{dp} \tag{5.104}$$

and

$$\frac{\partial \tilde{R}}{\partial \tilde{V}} = Q\frac{\partial R}{\partial \tilde{V}} = Q\,\frac{\partial R}{\partial V}Q^t \tag{5.105}$$

is the actual stiffness matrix of the Newton-Raphson procedure (see Eqs. (5.25) through (5.27)). Finally, Eq. (5.103) provides:

$$Q\frac{dV}{dp} = -\left(\frac{\partial \tilde{R}}{\partial \tilde{V}}\right)^{-1}\left(\frac{dQ}{dp}R + Q\left(\frac{\partial R}{\partial X}\frac{dX}{dp} + \frac{\partial R}{\partial p}\right)\right) = 0 \tag{5.106}$$

Therefore, it can be noticed that the derivatives are solutions of a linear problem, which is always the case in sensitivity analysis, and that the matrix is the same as the matrix of the direct problem, as the discrete differentiation method has been used. This significantly reduces the computational cost of the derivative calculations, because the matrix has already been triangularized. Actually, for nonlinear problems solved by the Newton-Raphson method, the current stiffness matrix is rather an approximation of $\partial \tilde{R}/(\partial \tilde{V})$ because it has not been computed with the exact velocity field (i.e. after the last iteration). However, practically it is quite satisfactory and does not justify new calculations.

Calculation of $\partial R/\partial X$

$\partial R/\partial p$ and dQ/dp are easily calculated while $\partial R/\partial X$ is more difficult to obtain although it does not present any specific theoretical difficulty. Its calculations are based on the mapping onto the reference element, as shown for the first term of Eq. (5.24):

$$\forall k,\ R_1^k(X,V) = \int_\Omega 2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-1}\dot{\varepsilon}_h{:}B^k\,dw_h \tag{5.107}$$

$$\text{so :}\qquad \forall k,\begin{cases} R_1^k(X,V) = \displaystyle\sum_e^{N\text{belt}} R_{1,e}^k(X,V) \\ \text{where:}\quad R_{1,e}^k(X,V) = \displaystyle\int_{\Omega_{\text{ref}}} 2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-1}\dot{\varepsilon}_h{:}B^k|J_e|\,dw_{\text{ref}} \end{cases} \tag{5.108}$$

where

Nbelt is the number of elements of the mesh,

Ω_{ref} is the reference element,

J_e is the jacobian matrix of the transform onto the reference element:

$$J_e = \frac{\partial x}{\partial x_{\text{ref}}} \tag{5.109}$$

and x_{ref} is the coordinate system of the reference element.

As Ω_{ref} is not dependent on the optimization parameters, the differentiation of Eq. (5.108b) can be easily carried out:

$$\begin{aligned} &\forall k,\ \forall e,\ \forall n, \\ &\frac{\partial R_{l,e}^k}{\partial X_n}(X,\mathrm{V}) = \int_{\Omega_{\text{ref}}} 2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-1}\dot{\varepsilon}_h{:}B^k\,\frac{\partial|J_e|}{\partial X_n}dw_{\text{ref}} \\ &\qquad + \int_{\Omega_{\text{ref}}} 2K(\sqrt{3}\dot{\bar{\varepsilon}}_h)^{m-3}\Big[2(m-1)\frac{\partial\dot{\varepsilon}_h}{\partial X_n}{:}B^k\dot{\varepsilon}_h{:}B^k \\ &\qquad + (\sqrt{3}\dot{\bar{\varepsilon}}_h)^2\Big(\frac{\partial\dot{\varepsilon}_h}{\partial X_n}{:}\ B^k + \dot{\varepsilon}_h{:}\frac{\partial B^k}{\partial X_n}\Big)\Big]|J_e|\,dw_{\text{ref}} \end{aligned} \tag{5.110}$$

The various derivatives of Eq. (5.110) are calculated in a standard way, as for other types of shape optimization problems. A precise description of these developments can be found in [15, 28], or in [70]. For more complex constitutive equations, the same approach can be followed, for instance as proposed in [46] and in [36]. The same kind of formulas are obtained for the other terms of the residual Eq. (5.24).

Therefore, this methodology allows us to compute dV/dp and then dX/dp at each time increment, and finally $d\Phi/dp$.

Accuracy of the Derivatives

As in structural mechanics, the derivative calculations are sensitive to the various time and space discretization errors. In fact, they are much more sensitive than the variables of the direct problem, as shown by the following examples. Thus, we shall see how the numerical approximations which are specific to

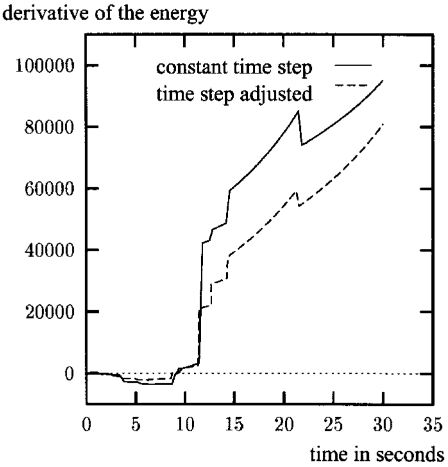


FIGURE 5.29 Time evolution of $d\Phi_{\text{ene}}/dp$ using either a constant time step or a time step adapted to the contact events.

the present forging model (time integration, contact evolution, mass conservation, remeshings) influence the derivatives and affect their accuracy.

Time Integration

We study the simple upsetting process described in Fig. 5.37. The contact is assumed to be perfectly sticking and the shape of the initial billet is described by a spline curve to be optimized. This academic problem requires a single forging step without contact evolution and without remeshing. It is well suited for testing the accuracy of the sensitivity analysis by comparing the analytical derivatives $d\Phi_{\text{EVE}}/dp$ of the total forging energy with the numerical values obtained by a finite difference scheme, as defined by Eqs. (5.4) and (5.5).

The estimated error on $d\Phi_{\text{EVE}}/dp$ is about 0.8 10^{-4}% at the first time increment. As the process simulation goes on, the accuracy decreases due to the cumulating of the discretization errors. Nevertheless, after 70 time increments, the error is still smaller than 0.1%, which is quite satisfactory, although this accuracy can be increased by reducing the time step size. In fact, it is noticed that, in logarithmic scale, the estimated error on dX/dp at the end of the upsetting process is a quasi-linear function of the time step size, as suggested by the discretization order of the time integration scheme (see Eq. (5.31)).

Contact Algorithm—Time Step Adaptation

We study again the academic forging example presented in Fig. 5.13. This is the forging of a simple cylinder by a die which shape is discretized by a three points spline. The shape sensitivity of the forming energy, $d\Phi_{\text{ene}}/dp$ evolves in time as shown in Fig. 5.29. It is initially equal to zero and continuously changes along with the material deformation and the nodal contact events. These events produce jumps in the sensitivity evolution, because when a node slightly penetrates the die, it is projected back onto the die surface (see Fig. 5.18). A more accurate strategy has been proposed by reducing the time step in order to avoid the penetrations (see Fig. 5.19). Figure 5.29 shows that $d\Phi_{\text{EVE}}/dp$ is strongly affected by the choice of the time step strategy, so the more accurate has to be used. Figure 5.30 shows that it provides a good agreement with the numerical derivatives: after a height reduction of nearly 40%, the difference between the two values of the objective function is about 0.5 10^{-5}%. Such a good result could not be obtained with a constant time step as Φ_{ene} was a discontinuous function of the die shape parameter p.

Incremental Mass Conservation

We previously described an incremental mass conservation formulation which makes it possible to avoid the numerical volume losses. In the present example, the classical formulation [Eq.(5.12)] provides a

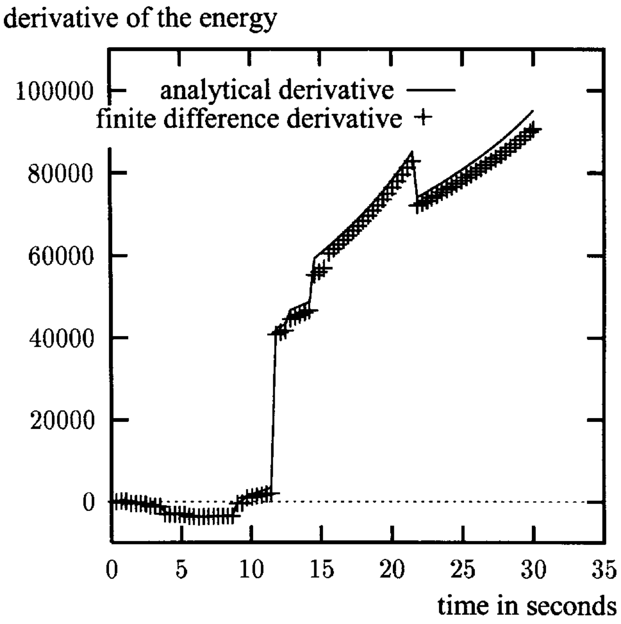


FIGURE 5.30 Time evolution of $d\Phi_{ene}/dp$ using a time step adapted to the contact events—comparison of the analytical and numerical derivatives.

reasonable volume loss of 0.4% at the end of forging. However, when it comes to the sensitivity analysis, this level of accuracy is not acceptable. In fact, Fig. 5.31 shows that in this case the time evolution of $d\Phi_{ene}/dp$ is almost negligible. This is explained by the numerical volume variations which are of the same magnitude as the variations due to the shape changes, and so hide the shape sensitivities. On the other hand, with the incremental formulation, the time evolution of the derivatives is quite satisfactory, as shown in Fig. 5.31.

Mesh Refinement

The same kind of phenomenon arises with respect to mesh refinement. It is well known that the finer the mesh, the better the finite element solution. However, numerical experiments show that the derivative calculations require more accurate solutions using finer meshes than the direct forming problem. For instance, in the extrusion example described (see Fig. 5.42), three different meshes have been tested (see Fig. 5.32). In Fig. 5.33, the extrusion force evolution is plotted for the three meshes. It shows that after 120 time increments a quasi-steady-state is reached. Due to discrete contact events, small oscillations can be noticed. Their frequency and amplitude depend on the mesh size, but, even for the coarser mesh, the forming force is estimated with a satisfactory accuracy. However, for the sensitivity analysis, Fig. 5.34 shows that the amplitude of the corresponding oscillations can be quite large. In fact, for the coarser mesh, it is not possible to conclude whether the derivative is positive or negative. With the intermediate mesh, the oscillations are reduced. Only the finer mesh makes it possible to estimate a steady-state value of the derivative.

This example shows the influence of the mesh size on the accuracy of the derivatives. For an actual non-steady-state forming process with remeshings, it is then also important to control the mesh size during the entire simulation. With the academic forging example presented in Fig. 5.13, three different values of the remeshing criteria have been tested. The corresponding meshes are shown in Fig 5.35 at the end of the simulations. Figure 5.36 shows the time evolution of $d\Phi_{ene}/dp$ for the different meshes. For the coarsest mesh, it seems that the derivatives are diverging after the second remeshing, while the finer meshes provide close solutions, so showing that the method is converging toward a given value of the derivative.

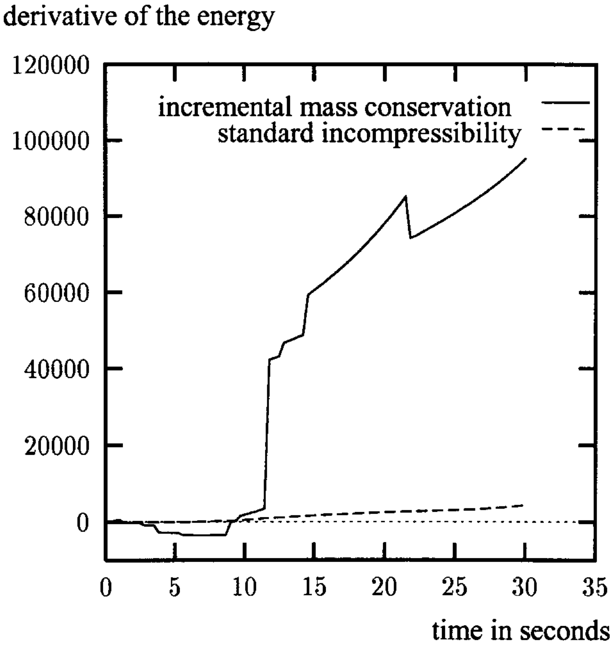


FIGURE 5.31 Time evolution of $d\Phi_{ene}/dp$ (analytical derivatives) using both the standard incompressibility equation [Eq. (5.25)] and the incremental mass conservation equation [Eq. (5.33)].

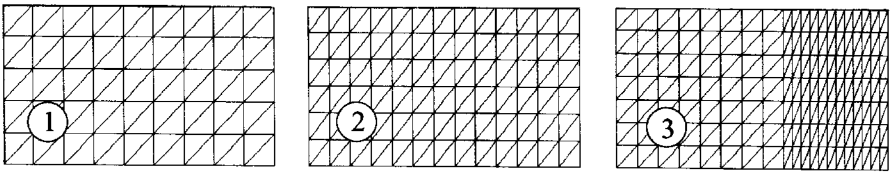


FIGURE 5.32 Three meshes of increasing refinement for the extrusion simulation.

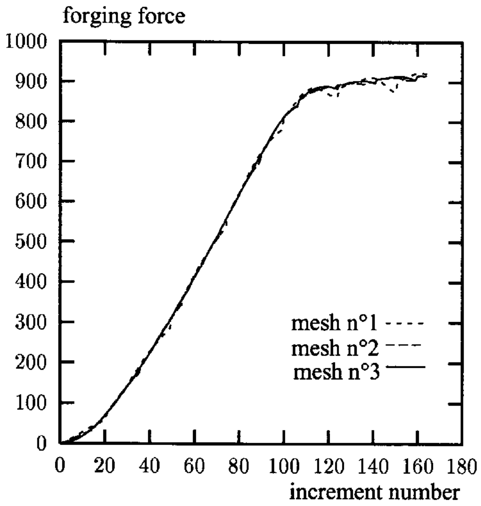


FIGURE 5.33 Time evolution of the forging force for the three different meshes: a quasi-steady-state is reached.

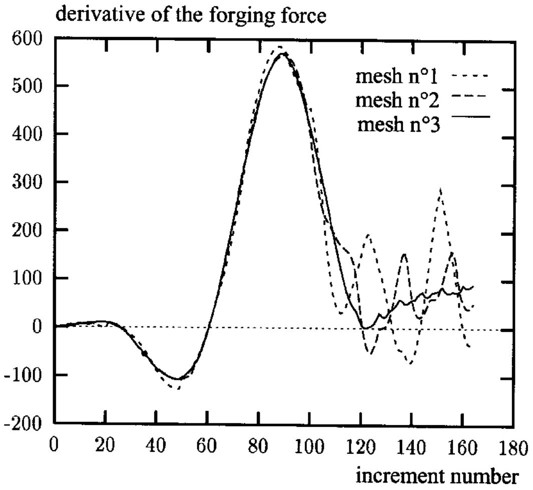


FIGURE 5.34 Time evolution of $d\Phi/dp$ (Φ being the extrusion force) for the three different meshes.

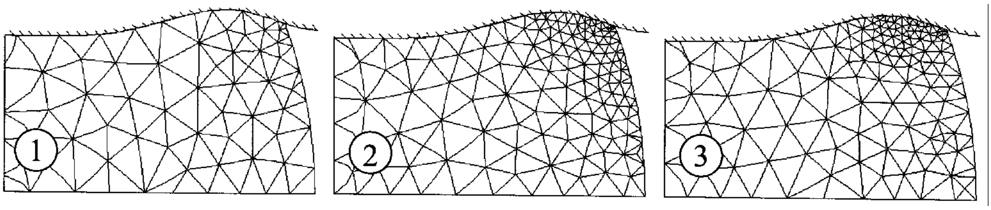


FIGURE 5.35 Three meshes obtained at the end of the academic forging example using three different values of the remeshing criteria.

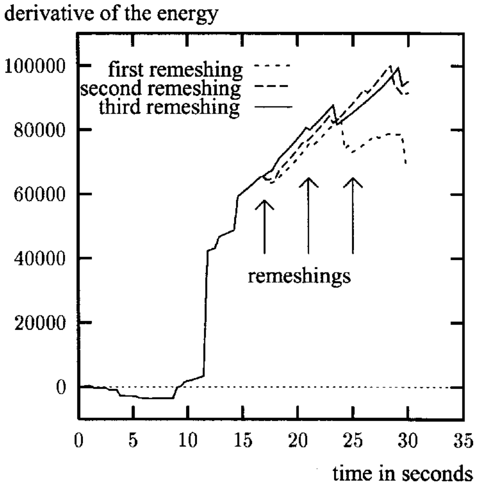


FIGURE 5.36 Time evolution of $d\Phi_{ene}/dp$ (analytical derivatives) for the three remeshing values.

Gradient-Based Optimization Algorithm

Several classical algorithms can be used for the minimization of the objective function Φ with respect to the shape parameters p. Quasi-Newton type algorithms are the most effective, and among them, the most widely used is the BFGS algorithm [20]. It consists of the following steps:

1. initialize p [choice of the initial shape and/or process parameters]
2. compute Φ, $d\Phi/dp$ [simulation of the successive operations of the forging sequence]
3. compute H_{BFGS}, a numerical approximation of the hessian matrix:

$$H_{BFGS} \approx \frac{d^2\Phi}{dp^2} \tag{5.111}$$

4. compute

$$\Delta P = -(H_{BFGS})^{-1}\frac{\partial\Phi}{\partial p} \tag{5.112}$$

5. update p into p', eventually using a line search procedure:

$$p' = p + \lambda\,\Delta p \tag{5.113}$$

where λ is the line search coefficient

6. check convergence; if verified stop; otherwise go to step 2.

H_{BFGS} is incrementally calculated as follows:

$$H_{i+1} = \left[I - \frac{\Delta P_i \Delta \mathrm{grad}(\Phi_i)^t}{\Delta P_i^t \Delta \mathrm{grad}(\Phi_i)}\right] H_i \left[I - \frac{\Delta \mathrm{grad}(\Phi_i)\Delta P_i^t}{\Delta P_i^t \Delta \mathrm{grad}(\Phi_i)}\right] + \frac{\Delta P_i \Delta P_i^t}{\Delta P_i^t \Delta \mathrm{grad}(\Phi_i)} \tag{5.114}$$

where

$$\Delta \mathrm{grad}(\Phi_i) = \mathrm{grad}(\Phi_{i+1}) - \mathrm{grad}(\Phi_i) \tag{5.115}$$

$$\Delta P_i = P_{i+1} - P_i \tag{5.116}$$

and I is the identity matrix

From a technical standpoint, the optimization software consists of three main modules. First, for any set of p values and the corresponding shapes, the forging simulation module, which is enriched with the objective function and gradient calculations, computes $\Phi(p)$ and $d\Phi/dp(p)$. These values are inputs for the optimization module which computes the new values of the shape parameters, p'. Finally, a conversion module generates the shapes of the forging dies, which correspond to the actual values of the parameters, in the format required by the simulation software.

5.5 Examples

Shape Optimization of the Initial Billet

After forging a straight cylinder, the free surface bulges (see Fig. 5.37). We look for the initial design of the billet which produces a straight side at the end of forging (see Fig. 5.38). First, the free surface of the billet is discretized by a spline curve. Then, the optimization algorithm is used to minimize the gap between the forged part and the desired cylinder. The Φ_{fil} function is used, without the 1_{desired} characteristic

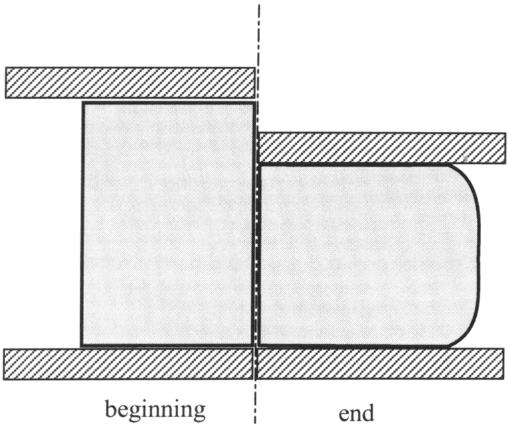


FIGURE 5.37 Upsetting of a straight cylinder—initial cylinder and final bulged part.

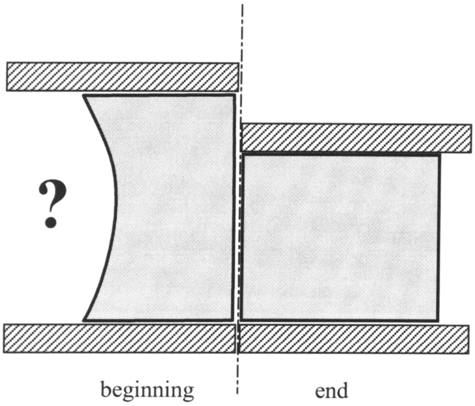


FIGURE 5.38 Initial billet shape optimization problem —expected shape of the final part.

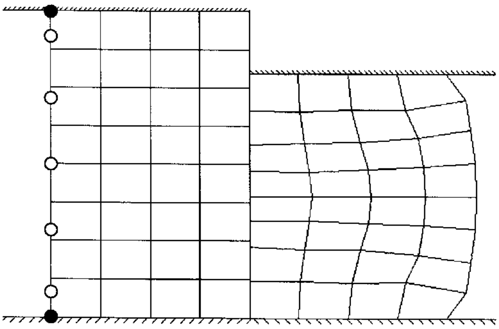


FIGURE 5.39 Upsetting of a straight cylinder—initial and final mesh—position of the characteristic spline points.

function. This rather academic example has been used by several authors [6, 25] to test the accuracy of the sensitivity analysis, and the convergence rate of the optimization algorithm.

The initial and final mesh configurations are shown in Fig. 5.39. The rheology is defined by: K = 4.5 kPa and $m = 0.15$. The contact is considered to be sticking in order to increase the bulging effect. The free surface of the billet, which is subjected to the optimization, is defined by a five points spline (see Fig. 5.39). The algorithm converges within 7 iterations which actually require 17 simulations due to the line search iterations. Although the iteration number is small, amazingly, numerical experiments show that the following more complex problems require even less iterations. This fact has also been noticed by other

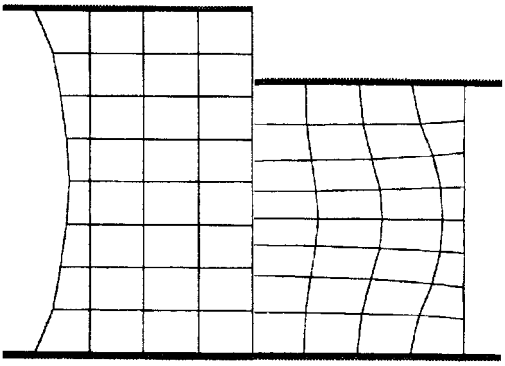


FIGURE 5.40 Optimal billet after 7 BFGS iterations and 17 simulations—Corresponding final part.

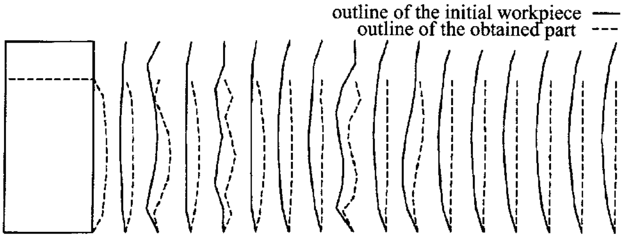

outline of the initial workpiece —
outline of the obtained part ---

FIGURE 5.41 Intermediate shapes of the free surface of the billet and the resulting obtained part, for each simulation during the optimization procedure.

authors using the same technique [5, 69]. The optimal preform shape is shown in Fig. 5.40, as well as the straight cylinder obtained at the end of the upsetting. The intermediate shapes which have been tested by the optimization procedure are shown in Fig. 5.41.

Shape Optimization of an Extrusion Die

Being a steady-state process, the design of optimal extrusion dies has been one of the first applications of shape optimization in metal forming [3, 13, 49]. The incremental sensitivity analysis developed here allows us to approach extrusion as a non steady-state process. The initial mesh and the extrusion dies are shown in Fig. 5.42. The materials behavior is viscoplastic. The simulation can be completed without remeshing. When friction is neglected and when the material behavior does not depend on temperature nor hardening, the process becomes steady as soon as the material front exits the die. This effect is shown by the isovalues of the strain rates (see Fig. 5.42). The following standard optimization problem can be proposed: find the shape of the die which minimizes the extrusion force. This shape is described by a spline, using two active shape parameters (see Fig. 5.42).

We have used this example to study the convergence behavior of the optimization algorithm. Actually, the graph of the objective function can be plotted, using a series of 80 simulations with different values of the two parameters and interpolating it (see Fig. 5.43). The function is convex and its unique minimum is easy to locate.

Figure 5.44 shows a 2D representation of the same plot with isovalues. In order to evaluate the influence of the initial guess on the convergence of the algorithm, we have chosen several starting points (circles in Fig. 5.44) randomly distributed in the space of the two parameters. For the seven tests, the shapes of the starting dies are shown in Fig. 5.45, along with the optimal die.

Whatever the initial design, the algorithm converges to almost the same minimum (triangles in Fig. 5.44) but with a different number of iterations. Table 5.1 shows the number of simulations needed for each test.

TABLE 5.1 Number of Optimization Iterations for the Different Starting Points

Case number	1	2	3	4	5	6	7
Number of extrusion simulations	37	21	39	23	41	19	1

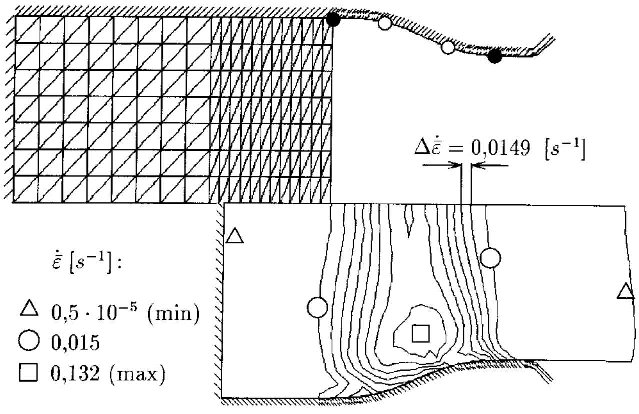

$\Delta\dot{\bar{\varepsilon}} = 0{,}0149\ [s^{-1}]$
$\dot{\bar{\varepsilon}}\ [s^{-1}]$:
△ $0{,}5 \cdot 10^{-5}$ (min)
○ 0,015
□ 0,132 (max)

FIGURE 5.42 Initial mesh and the extrusion die defined by a four points spline—Final mesh when a quasi steady-state has been reached and the iso-values of $\dot{\bar{\varepsilon}}$.

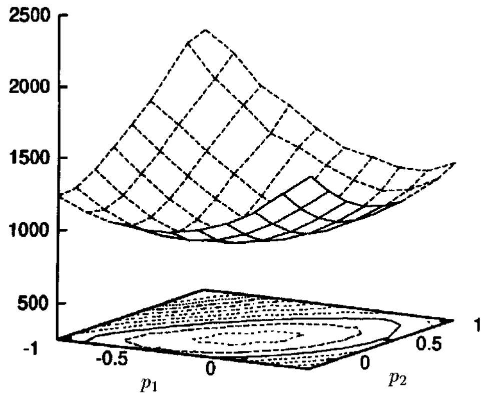


FIGURE 5.43 Plot of the objective function for the quasi steady-state extrusion problem.

The maximum gap between the different values of the objective function is less than 3 percent, which is close to the finite element discretization error for such a problem.

This shows the robustness of the optimization method, as well as its computational cost in terms of number of process simulations. However, as has just been mentioned, from the convergence rate standpoint, these examples are more severe than the forthcoming industrial applications.

Preform Shape Optimization for a Two-Step Forging Sequence

When the forging problem is not too different from problems handled in the past or when the shape of the final part is simple, experienced designers can find the appropriate general outline of the preforming dies. In these cases, the computer contribution will be to improve the proposed design, according to selected optimization criteria. In the present example, it is applied to the minimization of the total forming energy.

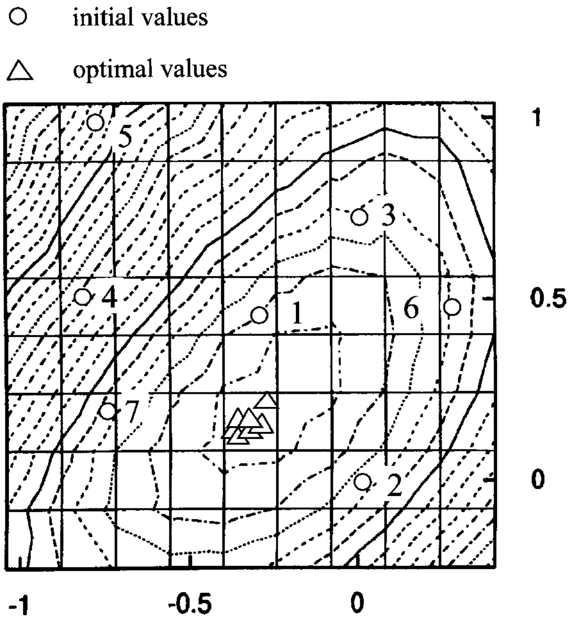


FIGURE 5.44 Position of the different initial values of the parameter (circles) and their respective convergence positions (triangles).

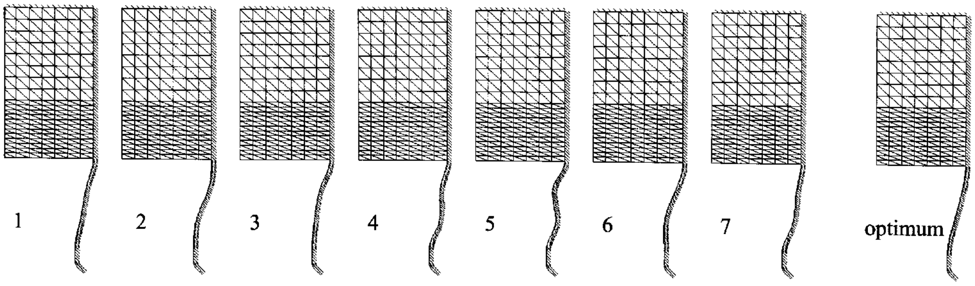
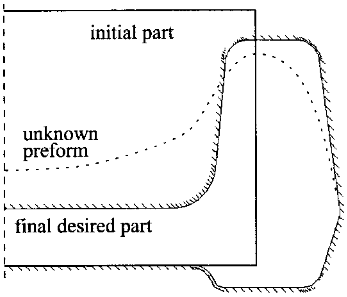


FIGURE 5.45 Different initial shapes which have been tested to find the same optimal die after different numbers of iterations.

Figure 5.46 shows ans axisymmetrical H-shaped part and the corresponding cylindrical billet. This kind of part is common in automobile, aeronautical, and other industries. It is relatively easy to find, by a heuristic trial-and-error technique or using the guidelines provided by technical manuals, a set of preforming dies that makes it possible to obtain the right final part at the end of forging (see Fig. 5.47). Therefore, the aim of the optimization is to optimize the preforming die shape in order to minimize the energy consumption during the complete forging process, while still ensuring the correct filling.

It has been decided to keep the same lower die during both operations, so only the upper die is subjected to optimization. It is defined using a cubic spline with five characteristic points. Three of them, represented in Fig. 5.47, are allowed to move their position. The Φ_{ene} objective function is selected and weighted by the Φ_{fil} constraint (the volume of the flash is not considered during its evaluation). The material is viscoplastic, with $m = 0.15$, and the friction coefficients are $q = 0.15$ and $\alpha = 0.2$. Each simulation of the forging sequence requires about 25 remeshings. The optimal shape is found after only four simulations, although five additional line search iterations have been carried out in order to ensure that the objective function cannot be further minimized. The optimal preforming die is shown in Fig. 5.48.

FIGURE 5.46 Preform shape optimization problem — Shape of the initial workpiece (cylinder), shape of the preform to be improved, and shape of the desired final part.



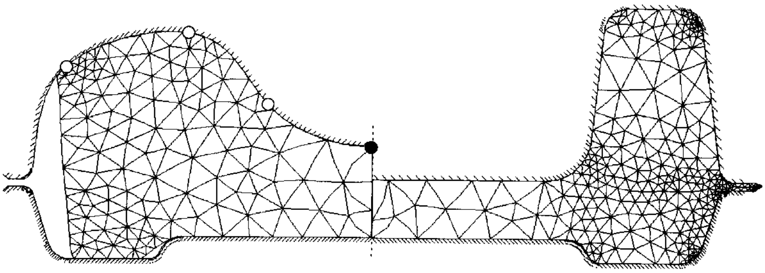


FIGURE 5.47 Initial forging sequence allowing us to obtain the right final part—Left side of the figure: end of the preforming operation and location of the spline active parameters—Right side of the figure: end of the finishing operation.

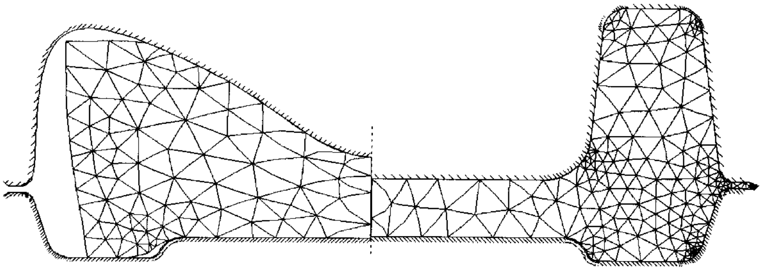


FIGURE 5.48 Optimal forging sequence minimizing the forming energy—Left side of the figure: end of the preforming operation and shape of the optimized die—Right side of the figure: end of the finishing operation.

The corresponding total forming energy is 13% smaller than these of the initial design. The forging force evolution with time is plotted in Fig. 5.49. It can be noticed that the optimized preform makes it possible to reduce by a factor of two the maximum forging force during the preforming operation. Also, during the finishing operation, the force is decreased but its maximum value is not changed as it is governed by the final required pressure to obtain a perfect filling.

In this case, the additional computational time due to the sensitivity analysis was less than 30% of the total simulation time, so showing the efficiency of the method. The optimization algorithm exhibited a fast convergence rate, while important changes of the die shape were produced, as shown in Fig. 5.50.

Preform Shape Design for a Two-Step Forging Sequence

The shape optimization method is not restricted to the optimization of an existing design but can also be applied to the preform tool design without preliminary information. Besides its scientific interest, this feature is very important, from a practical standpoint, when a completely new forging sequence has to

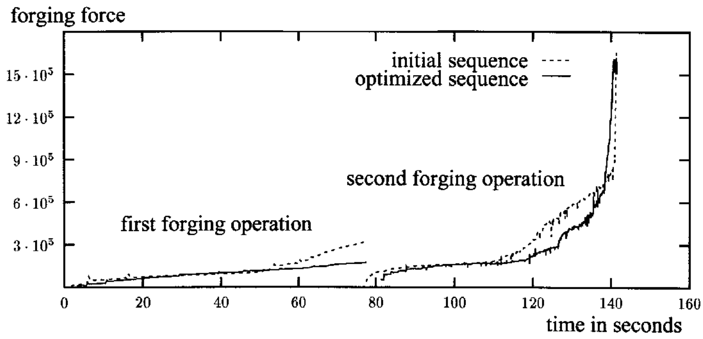


FIGURE 5.49 Forging force vs. time for the two forging operations, and for the initial and optimized designs.

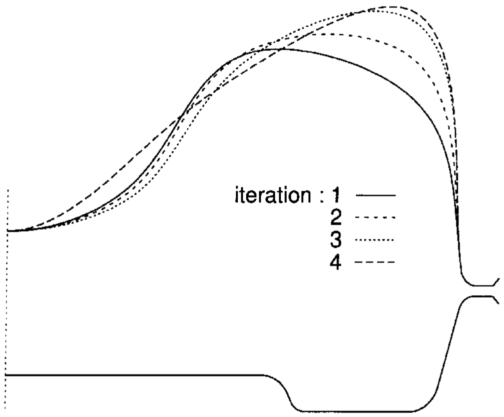


FIGURE 5.50 Successive tested shapes for the upper die of the preforming step.

be defined in a very short time, and from a more general standpoint as it makes it possible to enlarge the design strategies used in a company. Figure 5.51 shows a part which cannot be forged in a single operation with the proposed initial billet. So, an additional preforming step must be designed in order to improve the material flow and allow the perfect filling of the dies.

It is decided to carry out a simple rough design, the preform shape being described by a four points spline and three active parameters (see Fig. 5.52). The existing knowledge in the design of the preforming dies is not used, so we start the design procedure with a simple flat shape. This is of course a bad choices as can be seen in Fig 5.52. The objective function is the value of the unfilled volume, Φ_{fil}, and the volume of the flash is not considered. The forming energy, Φ_{ene}, is added as a regularization of the global objective function Φ_{reg}.

Only four simulations of the two step forging process were required to find a preforming die which provides a correct forming of the desired part. The final forging sequence is shown in Fig. 5.53. It can be noticed that the proposed preform shape is a smoothened sketch of the finishing die. It is rather smooth and easy to produce, and in agreement with the shapes classically obtained through knowledge-based approaches. The successive tested shapes during the optimization iterations are shown in Fig. 5.54. Figure 5.55 shows the evolution of the unfilled area during the automatic design. The unfilled area completely disappears at the fourth trial.

This example shows that, using this optimal design technique, a satisfactory design can be obtained by inexperienced people. The next example is an application to an actual industrial problem, the forging of the axisymmetrical rotor. The original design of the preforming tools (see Fig. 5.3) is not satisfactory as a folding defect occurs during the finishing operation (see Fig. 5.4). In fact, it is rather a piping defect which occurs under the inside rib due to a bad balance of different flows (see Fig. 5.5). Several trials and

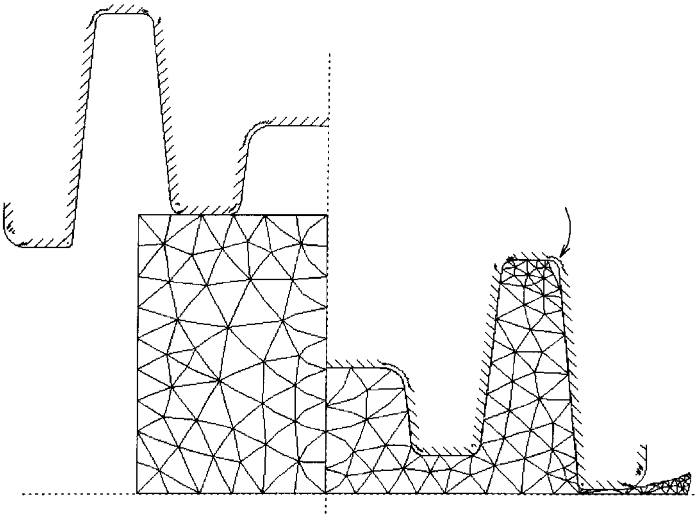


FIGURE 5.51 Attempt to forge the part in a single operation—Left side of the figure: initial workpiece between the forging dies—Right side of the figure: end of the forging operation, the upper part of the rib is not filled.

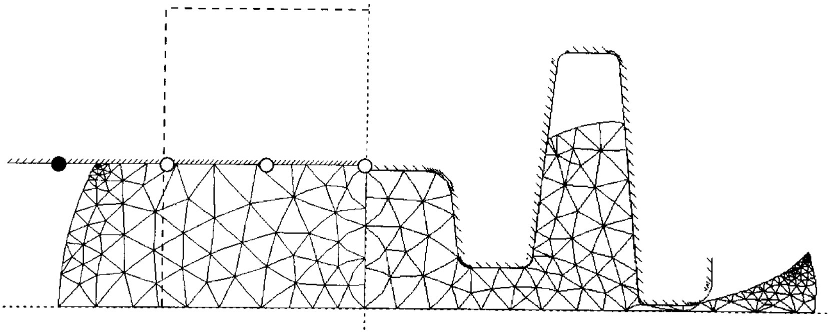


FIGURE 5.52 First iteration of the optimization procedure—Left side of the figure: end of the preforming operation, the preforming die being flat and defined by a four points spline, three of which are active (the dashed lines show the outline of the initial workpiece)—Right side of the figure: end of the finishing operation, the upper part of the rib is very badly filled.

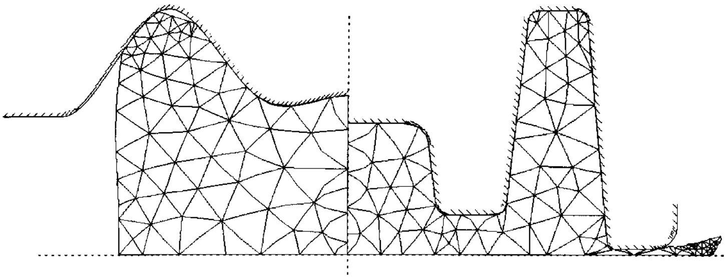


FIGURE 5.53 Fourth iteration of the optimization procedure—Left side of the figure: end of the preforming operation and shape of the optimized preforming die—Right side of the figure: end of the finishing operation, the desired part is perfectly obtained.

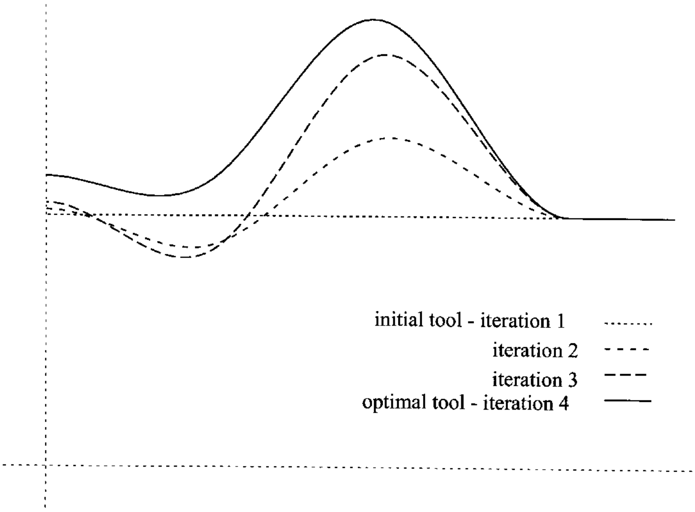


FIGURE 5.54 Successive shapes tested during the optimization procedure.

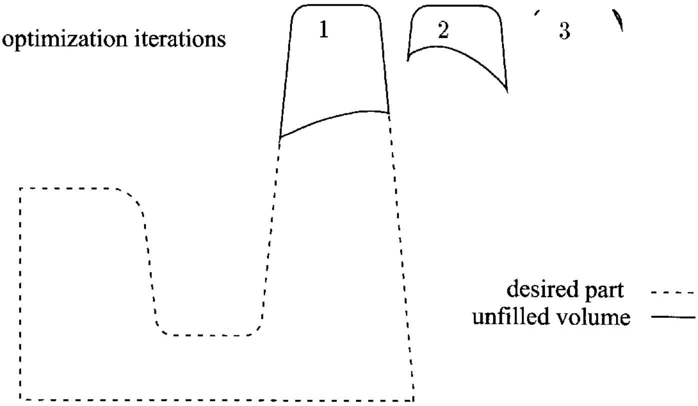


FIGURE 5.55 Evolution of the unfilled part of the rib during the optimization procedure.

errors have made it possible to suggest a satisfactory design of the preforming die (see Fig. 5.6) and it is interesting to contrast this result with the solution provided by the optimization method. Here, the selected objective function is Φ^2_{fold}. A part of the upper preform die is discretized by a seven points Bspline function while the other part of the shape and the lower die are kept constant. Initially, this new discretization is similar to the initial design for this example, and the piping defect occurs during the finishing operation. Three points of the Bspline are considered as active in the vertical direction for the optimization problem. Amazingly, at the second iteration of the procedure, the suggested preform shape (see Fig. 5.56) makes it possible to remove the piping defect and obtain the desired part at the end of the process (see Figs. 5.57 and 5.58). Figure 5.59 shows that the new proposed preform does not allow to prevent the concurrent flows which have caused the piping defects, but it rather makes it possible to avoid the initiation of this defect by a preliminary forging of this area.

This problem provides an actual example where the optimization method makes it possible to obtain a satisfactory design without much effort, while the manual design, though aided by the numerical simulation, is a tough and long process for inexperienced persons.

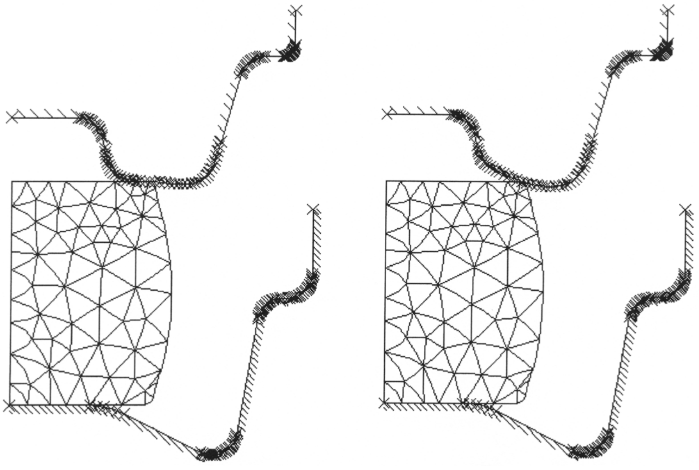


FIGURE 5.56 Initial (first trial of the optimization method) and new design (obtained by a single iteration of the optimization method) of the upper die of the preforming operation.

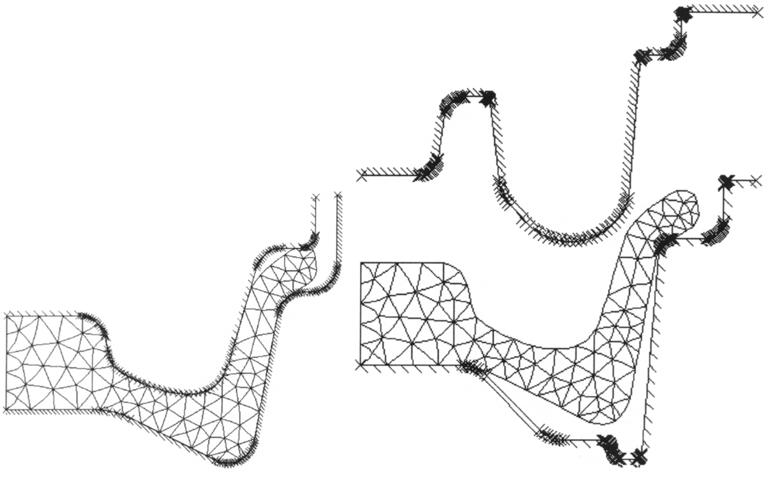


FIGURE 5.57 End of the preforming operation—New preform inside the finishing forging dies.

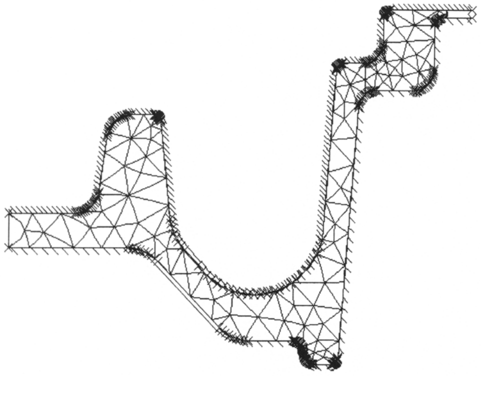


FIGURE 5.58 End of the finishing operation.

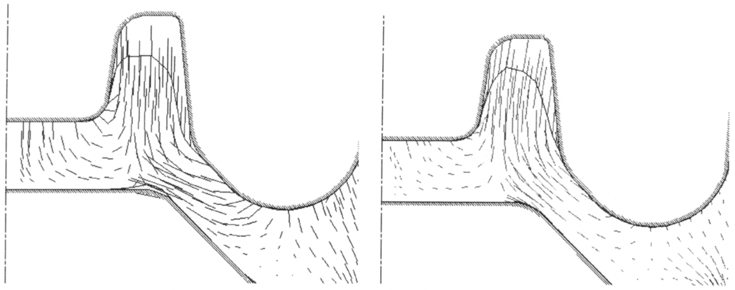


FIGURE 5.59 Comparison of the flows for the initial and new designs at an intermediate forging step of the last forming operation.

5.6 Conclusion

Optimal Design Techniques in Non-Steady-State Metal Forming Processes

We have seen that a wide variety of techniques have been investigated to develop computer aided design softwares in non-steady-state metal forming processes. They reflect two ways of approaching these issues: the methods based on artificial intelligence and the methods based on optimization algorithms. In fact, these two approaches are rather complementary. In a first step, an expert system, for instance, can provide an initial design to the forming problem. If the problem is new or different from the ones which have been previously studied, then, in a second step, numerical simulation softwares and automatic optimization techniques allow to improve this design, and possibly to suggest new ideas, even if the artificial intelligence based methods have failed. For this reason, this presentation has been focused on optimization methods because they allow to both solve more complex problems and to increase the knowledge.

In short, optimization methods are approached in two ways: approximating the objective functions or analytically computing their gradients. The first method is much easier to develop and to apply to any simulation software. However, it is expensive in terms of computational time. Consequently, it is rather restricted to problems with few parameters and limited variations of these parameters. On the other hand, optimization methods using the exact gradients calculations provide much faster results, but require longer development times. As these calculations are complex, simplification assumptions are often introduced, for instance by neglecting the thermal effects. However, recent results show how effective these methods can be. They make it possible to design reasonable forging sequences starting from crash, although several difficulties are faced, mostly connected with the accuracy of the finite element method. An important aspect is that, for problems with remeshings, the gradients cannot be computed by numerical differences, while this method is easy to implement (although computationally expensive). Indeed, in the frame of non-steady-state processes, it is quite inaccurate. The other important aspect is that the gradients are sensitive to the space and time discretization errors, much more sensitive than the variables of the direct problem (the simulation of the forming process).

To sum up, it is believed that these two optimization methods are complementary. Using simplification assumptions, the methods based on the gradient calculations can be used to suggest a first rather satisfactory design. It makes it possible to reduce the number of optimization parameters and their range of variation, so the methods based on the approximation of the objective functions can be used to improve this design, without requiring the previous simplification assumptions.

Finally, we have noticed that optimization techniques for non-steady-state forming processes are a rather new field of research. This late start is easily explained by the complexity of the processes and more particularly by the requirements of accuracy on the numerical solution. However, the obtained results are encouraging and show that in a near future such techniques will be quite helpful in metal forming. The presented examples show that they can already solve complex forming problems, such as designing the preforming tools of a complex forging sequence, making it possible either to remove folding defects, to obtain the right forged part at the end of the process, or to minimize the forging energy. The proposed new design methodologies introduce new ideas and may allow standardization of the design procedures.

Prospect

The future of the optimization techniques is rich. First, more complex constitutive equations, more complex metal forming problems, and more complex objective functions can be handled with the same approach. It requires some additional efforts for their developments but does not present any theoretical difficulty. The multi-objective optimization has not been much studied till now but several strategies can be developed in order to answer more precisely the actual design problem, which is always a compromise between several objectives. Our presentation has been limited to axisymmetrical problems, mainly for computational time reasons. However, three-dimensional non-steady-state metal forming softwares are available, exhibiting the same level of accuracy as two-dimensional softwares. It is then possible to extend these techniques to three-dimensional processes. Moreover, it can be noticed that several forging preforms are axisymmetrical while the final part is three-dimensional. These problems provide an intermediate step before handling a complete three-dimensional optimization procedure and the related computational costs. Finally, we have seen that these methods can be applied to a wide range of metal forming processes, but they can also be extended to other materials such as polymers, glass, etc. For instance, they could be applied to the thermal regulation of the extrusion dies of polymer profiles, or to the pressure cycle optimization of the glass bottle blowing process.

Regarding the development these techniques, two different costs have been emphasized: the computational cost of the simulation and the human cost required by the differentiation of the finite element softwares. For the first issue, the generalization of parallel computer offers an easy answer. The optimization method based on the approximation of the objective functions is easy to parallelize with a high efficiency. For the methods based on the gradients calculations, the parallelization is tougher. It requires the parallelization of the metal forming softwares. However, this is already available for some. Regarding the second source of costs, a lot of potentials are offered by the automatic differentiation softwares. In fact, they make it possible to carry out a fast and reliable sensitivity analysis. The generated code is rather inefficient in terms of computational time and it is too memory requiring. Consequently, it has to be improved and optimized by hand. This is, however, much easier and much faster than carrying out the entire differentiation by hand. So, an important reduction of the development time is expected from this technique.

For the closure, we believe that in a near future optimization methods will be as popular in metal forming applications as they are now in structural mechanics.

References

1. Ahlberg, J.H., Nilson, E.N., and Walsh, J.L., *The theory of splines and their applications, Série Mathematics in science and engineering,* vol. 38, Academic Press, London, 1967.
2. Altan, A., Boulger, F.W., Becker, J.R., Akgerman, N., and Henning, H.J., *Forging equipment, materials and practice,* National Technical Information Service, U.S. Department of Commerce, Springfield, Virginia, October (1973)
3. Antúnez, H.J., Kleiber, M., and Sosnowski, W., *Shape and non-shape sensitivity and optimization of metal forming processes, Computational Plasticity-Fundamental and Applications,* Proceedings of the Fifth International Conference, Barcelona, 17–20 March, p 783, D.R.J. Owen et al. éditeurs, Pineridge Press, Swansea, UK, 1997.
4. Anza, J.J., Barbero, J.I., Muga, C., Calvo, J., Carillo, A.R., and de Pedro, A.R., Computer aided forging design, *J. Mater. Process. Technol.,* 45:715, 1994.
5. Badrinarayanan, S., Constantinescu, A., Zabaras, N., Preform design in metal forming, Proceedings of the 5th International conference on Numerical Methods in Industrial Forming Processes, 18–21 June, Ithaca, New York, Balkema, Rotterdam, 533, 1995.

6. Badrinarayanan, S., Zabaras, N., Sensitivity analysis for the optimal design of forming processes, *Comp. Meth. Appl. Mech. Eng.,* 129:4:319, Feb. 1996.
7. Bathe, K.J., *Finite element procedures,* Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1996.
8. Beckers, M., and Kopp, R., *A new approach to optimization of metal forming processes,* Thompson et al. (editors), Numiform 89, 107-113, Balkema, 1989.
9. Bohatier, C., and Chenot, J.L., Finite element formulation for non-steady state large viscoplastic deformation, *Int. J. Num. Meth. Eng.,* 21:1697, 1985.
10. Brezzi, F., *On the existence, uniqueness and approximation of saddle point problems arising from lagrangian multipliers,* R.A.I.R.O., 8ième année, R.2 p. 129, 1974.
11. Chamouard, A., Statique et dynamique appliquée au déformations à chaud, *Estampage & Froge,* vol. 1 and 2, Dunod editor, 1966. (in French)
12. Chenot, J.L., Coupez, T., and Fourment , L., Recent progresses in finite element simulation of the forging process, Computational Plasticity-Fundamental and Applications, Proceedings of the Fourth International Conference, Barcelona, 3-6 avril, p. 1321, D.R.J. Owen et al. éditeurs, Pineridge Press, Swansea, UK, 1995.
13. Chen, M.F and Maniatty, A.M., An inverse technique for the optimization of some forming processes, Proceedings of the 5th International conference on Numerical Methods in Industrial Forming Processes, 18-21 June, Ithaca, New York, Balkema, Rotterdam, 545, 1995.
14. Chenot, J.L., Massoni, E., Fourment, L., Inverse problems in finite element simulation of metal forming processes, Engineering Computations, vol. 13, n°2/3/4, p.190, 1996.
15. Chenot, J.L., Coupez, T., Fourment, L., and Ducloux, R., 3-D finite element simulation of the forging process: scientific developments and industrial applications, Advanced Technology of Plasticity 1996, T. Altan editor, Proceedings of the 5th International Conference on Technology of Plasticity, October 7-11, Colombus, 465, 1996.
16. Choi, S.H. and Dean, T.A., Computer aided design of die block layout, *Int. J. Mach. Tools Manufact.,* 27(1): 91, 1987
17. Chu, K. and Lee, D., Computer aided analysis of sheet metal forming processes, Proceedings of the First International Conference on Technology of Plasticity, published by the Japan Society for Technology of Plasticity and the Japan Society of Precision Engineering, Tokyo, 1: 660, 1984.
18. Coupez, T. and Chenot, J.L., Large deformations and automatic remeshing, Computational Plasticity, Pineridge Press, edited by Owen, Oñate, Hinton, Pineridge Press, Swansea, 1077, 1992.
19. Cox, M.G., The numerical evaluation of B-splines, National Physical Laboratory DNAV, August 1971.
20. Dennis, J.E. and Schnabel, R.B., *Numerical methods for unconstrained optimization and non linear equations,* Prentice-Hall, Englewood Cliffs, New Jersey, 1983.
21. Duggirala, R., Shivpuri, R., Kini, S., Ghosh, S., and Roy, S., Computer Aided Approach for Design and Optimization of Cold Forging Sequences for Automotive Parts, *J. of Mater. Proc. Technol.* 46:185, 1994.
22. Espiga, F., Jugo, A., and Anza, J.J., Industrial application of the numerical simulation to the design and optimization of forging processes, *J. Mater. Process. Technol.,* 45: 81, 1994.
23. Felder, E., The wear of hot working tools. Application to forging and rolling of steel, The tribological design of machines and elements, D. Dowson et al. (editors), Trib. Series 14, Elsevier, 253, 1989.
24. Fonseca, C.M. and Fleming, P.J., Genetic algorithms for multiobjective optimization : formulation, discussion and generalization, Proc. Of the Fifth Int. Conf. On Genetic Algorithms, Morgan-Kauffman (editors), 416, 1993.
25. Fourment, L., Chenot, J.L., The inverse problem of design in forging, Inverse Problems in Engineering Mechanics, Balkema, Rotterdam, 21, 1994.
26. Fourment, L., Balan, T., and Chenot, J.L. Shape optimal design in forging, Proceedings of the 5th International conference on Numerical Methods in Industrial Forming Processes, 18-21 June, Ithaca, New York, Balkema, Rotterdam, 557, 1995.
27. Fourment, L., Miles, M.P., Chenot J.L., Incremental mass conservation and adaptive remeshing for the thermo-mechanical coupling between workpiece and tools in nonsteady-state metal forming,

Proceedings of the 5th International conference on Numerical Methods in Industrial Forming Processes, 18-21 June, Ithaca, New York, Balkema, Rotterdam, 431, 1995.

28. Fourment, L. and Chenot J.L., Optimal design for non steady-state metal forming processes- I Shape optimization method, International Journal for Numerical Methods in Engineering, 39: 1: 33, 1996.
29. Fourment, L., Balan, T., and Chenot J.L., Optimal design for non steady-state metal forming processes- II Application of shape optimization in forging, International Journal for Numerical Methods in Engineering, vol. 39: 1: 51, 1996.
30. Fourment, L., Balan, T., and Chenot, J.L., Shape optimization of the preforming tools in forging, Proceedings of the 2nd International Conference on Inverse Problems in Engineering: theory and practice, June 9-14, Le Croisic, 1996.
31. Fourment, L., Balan, T., and Chenot, J.L., A software for preforming tools optimization in forging, Advanced Technology of Plasticity 1996, T. Altan editor, Proceedings of the 5th International Conference on Technology of Plasticity, October 7-11, Colombus, 449, 1996.
32. Fourment, L., Balan, T., Chenot, J.L. Optimum design of the hot forging process: a F.E. inverse model with remeshing for large deformations, Computational Plasticity - Fundamental and Applications, Proceedings of the Fifth International Conference, Barcelona, 17-20 March, p. 804. D.R.J. Owen et al. éditeurs, Pineridge Press, Swansea, UK, 1997.
33. Garcia S., and Scott, E.P., Use of genetic algorithms in optimal experimental desigs, Proceedings of the 2nd International Conference on Inverse Problems in Engineering: theory and practice, June 9-14, Le Croisic, 1996.
34. Germain, Y., Chenot, J.L., and Mosser, P.E., Finite element analysis of shaped lead-tin disk forging, Proc. of the Numiform'86 conference, Gothenbourg, Mattiasson et al. Editors, 271, 1986.
35. Ghouati, O. and Gelin, J.C., Inverse identification methods for material parameters estimation in metal forming, Proceedings of the 5th International conference on Numerical Methods in Industrial Forming Processes, 18–21 June, Ithaca, New York, Balkema, Rotterdam, 569, 1995.
36. Ghouati, O. and Gelin, J.C., Sensitivity analysis and optimization of shape and process parameters in metal forming, PD-Vol 75, Engineering Systems Designs and Analysis, 3: 221, 1996.
37. Ghouali, A., Ortola, S., Oster, A., and Duvaut, G., Local sensitivity analysis for the forging gamut optimization, Proc. of the Colloque IDMME'96, Nantes (France), 985, 1996.
38. Glynn, D., Lyons, G., and Monaghan, J., Forging sequence design using expert system, Int. J. Of Mater. Proc. Technology, 55: 95, 1995.
39. Hafka, R.T., Gürdal, Z., Kamat, M.P., Element of structural optimization, Kluwer Academic Publishers, 1990.
40. Hans Raj, K., Fourment, L., Coupez, T., and Chenot, J.L., Simulation of industrial forging of axisymmetrical parts, Engineering Computations, 9: 575, 1992.
41. Han, C.S., Grandhi, R.V., and Srinivasan, R., Optimum design of forging die shapes using nonlinear finite element analysis, AIAA Journal, 31: 4: 774, April 1993.
42. Harnaes, J. and Wanheim, T., Simulation, of cold forging folding problems using model materials, *J. of Mech. Working Tech.,* 12: 126, 1985.
43. Hwang, S.M. and Kobayashi, S., Preform design in plane-strain rolling by the finite element method, *Int. J. Tool Des Res.,* 24: 4: 253, 1984.
44. Kang, B.S. and Kobayashi, S., Preform design in ring rolling processes by the three-dimensional finite element method, *Int. J. Tool Des Res.,* 139, 1991.
45. Kim, D. J. Kim, B.M., and Choi, J.C., Determination of optimal billet size using the artificial neural networks and the finite element method for a forged product, Advanced Technology of Plasticity 1996, T. Altan editor, Proceedings of the 5th International Conference on Technology of Plasticity, October 7-11, Colombus, 299, 1996.
46. Kleiber, M., Hien, T.D., Antúnez, H., and Kowalczyk, P., Parameter sensitivity of elastoplastic response, *Engineering Computations,* 12: 263, 1995.
47. Kobayashi, S. Oh, S., and Altan, T., *Metal forming and the finite element method,* Oxford University Press, 1989.

48. Krishnakumar, K. Micro-genetic algorithms for stationary and non-stationary function optimization, SPIE-Intelligent Control and Adaptive Systems, 1196, 289, 1989.
49. Kusiak, J. Thompson, E.G., Optimization techniques for extrusion die shape design, Numerical Method in Industrial Forming Processes, edited by Mattiason, Samuelson, Wood, Zienkiewicz, Balkema/Rotterdam/Boston, 569, 1989.
50. Kusiak, J., A technique of tool-shape optimization in large scale problems of metal forming, *J. of Mater. Processing Technol.,* 57: 79, 1996.
51. Lange, K., et al., *Handbook of metal forming,* Mc Graw Hill, New York, 1985.
52. Le Floc'h, A., Wey, E., Jacomet, S., Poitrault, I., Pisseloup, J., and Bocquet, P., Deformations estimate of an open-die forged heavy disc and comparison with an industrial one, Physical Modeling of Metalworking Processes, E.Erman, S.L. Semiatin (editors), The Metallurgical Society, Warrendale, Pennsylvania, 81, 1987.
53. Lengyel, B., Chaudhry, I.A., Hibberd, R.D., A consultative system for cold forging with finite element predictions, *J. Mater. Process. Technol.,* 45: 93, 1994.
54. Liszka, T. and Orkisz, J., The finite difference method at arbitrary irregular grids and its application in applied mechanics, *Comp. and Struct.,* 11: 83, 1980.
55. Maniatty, A.M. and Chen, M.F., Shape sensitivity analysis for steady metal-forming processes, *International Journal of Numerical Methods in Engineering,* 39: 7: 1199, 1996.
56. Marin, P. Lee, C., Tichkewicz, S., Tacking into account thermal effects in ForgeRond, a fast hot-forging simulation tool, Advanced Technology of Plasticity 1996, T. Altan editor, Proceedings of the 5th International Conference on Technology of Plasticity, October 7-11, Colombus, 475, 1996.
57. Park, J.J., Rebelo, N., and Kobayashi, S., A new approach to preform design in metal forming with the finite element method, *Int. J. Tool Des Res.,* 23: 1: 71, 1983.
58. Pironneau, O., Optimal shape design for elliptic systems, Springer Verlag, 1984.
59. Roy, R., Chodnikiewicz, K., Balendra, R., Interpolation of forging preform shapes using neural networks, *J. Mater. Process. Technol.,* 45: 694, 1994.
60. Santos, M.T., Jugos, A., Ojanguren, J.M., and Lama, J., Aplicación de técnicas de diseño de experimentos en l optimización del diseño de procesos industriales de forja, proceedings of the XII Congreso Nacional de Ingeniería Mecánica, 505, 1997.
61. Smelser, R.E., Richmond, O., Thompson, E.G., A numerical study of the effect of die profile on extrusion, Numerical Method in Industrial Forming Processes, edited by Mattiason, Samuelson, Wood, Zienkiewicz, Balkema/Rotterdam/Boston, 305, 1986.
62. Tikhonov, A., Arsine, V., Solution of ill-posed problems, V.H. Hinston and sons, Washington, 1977.
63. Tortoreli, D.A. and Micharelis, P., Design sensitivity analysis : overview and review, Inverse Problems in Engineering, 1:71, 1994.
64. Wanheim, T., Simulation with model material in the nordic countries : projet SIMON, *J. of Mech. Working Tech.,* 12: 135, 1985.
65. Wu, W.T., Oh, S.I., Altan, T., Investigation of defect formation in rib-web type forging by ALPID, Proc. of NAMRC XII, Michigan Technological University, Houghton, Michigan, 1984.
66. Wu, W.T., Li, G.J., Arvind, A., and Tang, J.P., 3D finite element analysis of industrial metal forming processes, Advanced Technology of Plasticity 1996, T. Altan editor, Proceedings of the 5th International Conference on Technology of Plasticity, October 7-11, Colombus, 479, 1996.
67. Wybo, J.L., Geffraye, F., Russeil, A., PROFIL: a decision support tool for metallic sections design using a CBR approach, Lecture notes in Artificial Intelligence 1010, Veloso.Aamodt (Eds.), Springer, October, 33, 1995.
68. Zhao, G., Wright, E., Grandhi, R.V., Forging preform design with shape complexity control in simulating backward deformation, *Int. J. Mach. Tools Manufact.,* 35(9), 1225, 1995.
69. Zhao, G., Huff, R., Hutter, A., Grandhi, R.V., Sensitivity analysis based preform die shape design using the finite element method, *J. Materials Engineering and Performance,* 6: 303, 1997.
70. Zhao, G., Wright, E., Grandhi, R.V., Preform die shape design in metal forming using an optimization method, *Int. J. Num. Meth. Eng.,* (in press) 1997.

6

Analysis of Welds Using Geometric Constraints

6.1 Introduction
6.2 Physical Model of Welding Process
6.3 Discussion of Mathematical Issues
6.4 Theoretical Foundation of the Geometric-Constraints Method
6.5 Some Properties Associated with Geometric Constraints
6.6 Prototype Case Study Analyses of Welds
6.7 Implicit Representation of Temperature Dependent of Material Properties
6.8 Implicit Representation of Coupling Between Heat Transfer and Fluid Convection
6.9 Conclusion

S.G. Lambrakos
Naval Research Laboratory

D.W. Moon
Naval Research Laboratory

We present a series of prototype analyses of weld morphology which typically occurs in a range of different types of welding processes ranging from Gas Metal Arc welding of steel to deep penetration welding processes using electron or laser beams. Our method of analysis is based on geometric constraints whose underlying properties are discussed in this report in terms of a general survey. Our description of this method is in terms of its formal mathematical foundation which is within the theory of constrained optimization. The formulation and prototype analyses presented in this report serve as a general outline for the extension and application of the geometric-constraints method to the analysis of complex weld morphology. This includes the analysis of welds where there can be fragmented and incomplete information concerning material properties, complex shapes of the observed solidification boundary (and associated heat-affected-zone), and only approximate information concerning the character of energy and filler metal deposition.

6.1 Introduction

In a series of reports[1,2] we presented a general and flexible approach for the numerical modeling and analysis of dynamic welding processes using a geometric-constraints method which provides a general framework for the inclusion of boundary value information easily obtained from experimental measurements or information based on assumptions concerning the physical character of the welding process. An important feature of the procedure for imposing constraints according to boundary value information is that it tends to compensate for either the unavailability of difficult experimental measurements (of material properties) or gaps in knowledge concerning the characteristics of the energy source. In References 1 and 2 the geometric-constraints method was described in terms of its application for the analysis of deep penetration welding processes.

In this report we present application of the geometric-constraints method to the modeling and analysis of welds which typically result from a range of different types of welding processes, including gas metal arc (GMA) and shielded metal arc (SMA) or gas tungsten arc (GTA), consisting of either single or multiple passes. We present a series of prototype case studies of welds typical of both bead-on-plate and bead-in-groove welding of stainless steel. An interesting feature of our approach is that it represents an extension of the method of distributed heat sources.[3] This method has been shown to be well-posed for providing an approximate representation of GMA welding processes where the combined influence of arc energy distribution and filler-metal additions can result in relatively complex shapes of the transverse cross section of the weldment. In particular, these complex shapes include the formation of weld crater/weld finger shapes, or in general, complex non-semi elliptical shapes, which are observed frequently in GMA weldments. The geometric-constraints method presented in this report represents a relatively more quantitative approach to simulating the type of weldments which typically result from single and multipass GMA welding processes than methods where the solution is driven primarily by a source term in the energy equation. We consider a series of cases, giving examples of analysis where different types of information concerning a given weld are assumed to be known. We attempt to examine the practical issue of what is to be considered sufficient information for assuring a reasonable level of accuracy for the solution and its effective uniqueness when *a priori* information about a given weld is either incomplete or fragmented.

Our presentation of the mathematical foundation of the geometric constraints method, which is in terms of constrained optimization theory,[4,5,6] and of the associated mathematical properties of pseudo-steady-state temperature fields, is relatively complete. It is not, however, rigorous. Our presentation is for the most part in terms of general illustrations of the underlying mathematical concepts rather than a rigorous mathematical development. In particular, our discussion of the mathematical properties of the steady state temperature field are in terms of either continuous or discrete field representations, depending upon which type of representation provides a more convenient description of a given property.

6.2 Physical Model of Welding Process

The model system to be specified is that of the temperature field of a dynamic weld in the pseudo steady-state. That is to say, the model system is characterized by quasi-steady energy transport in a coordinate system that is fixed in the reference frame of a moving energy source. The outermost boundaries of the model system are defined by the sides of a finite-sized rectangular region. Although the model system is formally that of a general time dependent system, only the pseudo steady-state solution is of relevance with respect to geometric-constraint information. In our development, the time step and volumetric discretization of the system serve as a means of constructing a weighted sum for the purpose of optimizing the discrete temperature field corresponding to a given set of geometric constraints. The input energy source is effected via specification of an upstream boundary condition on the model system. The equations governing the model system are

$$\rho(T)C_P(T)\frac{\partial T}{\partial t} = \nabla \cdot [k(T)\nabla T] + \nabla \cdot q(T)\nabla \cdot q_s(x) \tag{6.1}$$

where the quantities $\rho(T)$, $C_p(T)$, and $k(T)$ are the temperature dependent density, heat capacity, and conductivity, respectively, and

$$\int_t^{t+\tau} \nabla \cdot q(T)dt = \Delta H \tag{6.2}$$

where the quantity ΔH is the change in energy per unit volume due to the temperature dependent source term $\nabla \cdot q\ (T)$. The source term $\nabla \cdot q_s\ (x)$ is associated with the input of energy into the system resulting from an external energy source. For the case study analysis given in this report this term is not adopted explicitly as a source term, but rather as a generating function for construction of an upstream boundary whose values serve as a source condition for the input of energy into the system. The geometric constraints method, however, can be adapted for the inclusion of information concerning the detailed character of an energy source if this type of information is available. This issue is discussed below. In this report the temperature dependent source term $\nabla \cdot q\ (T)$ is considered implicitly via specification of an effective diffusivity $\kappa\ (T)$. This method of representing the influence of temperature-dependent source terms is convenient for inclusion into model representations which do not consider time explicitly, e.g., the dynamic pseudo-steady-state temperature field of a weld that would occur during a continuous welding process.

6.3 Discussion of Mathematical Issues

The geometric constraints method as applied in this study represents, in a particular sense, a generalized extension of the method of distributed heat sources.[3] The method of distributed heat sources entails constructing a two-dimensional distribution of heat sources consisting of a finite set of moving point sources located upon a plane which is located upstream with respect to the trailing three-dimensional solidification boundary. Following this method, the strengths and relative positions of the point sources are adjusted according to the transverse cross section of the weld melt pool. For a given set of material properties and a specified value of the total heat input per unit distance along the direction of relative motion between the energy source and workpiece, it can be shown that there is a unique one-to-one mapping between any two-dimensional temperature distribution upstream, in the neighborhood of the heat sources, and the trailing three-dimensional solidification boundary, downstream. It is this property of a closed and bounded solution domain[7,8] which provides a basis for construction of an accurate solution via the method of distributed heat sources. There is an inherent limitation on the accuracy of this method because of the conditions for its application in practice. The general approach underlying this method, however, can be generalized by methods of constrained optimization to include a wide range of different types of information concerning the weld properties and process parameters.

In practice, one could be faced with a range of incomplete information concerning the welding process and with characteristics of the pseudo-steady-state temperature field of the weld which are not conveniently represented mathematically via analytical formulations defined for a closed and bounded solution domain having a relatively simple shape. Among the issues to be considered for modelling a weld system are the following:

1. The total energy input per unit distance for a given welding process may only be known approximately. The energy input per unit distance represents a global constraint on the three-dimension temperature field for the pseudo-steady-state of the weld. For cases where this global constraint on the system is not specified accurately, uniqueness cannot be assured for temperature fields calculated according to a method based on distributed heat sources whose relative positions, upstream, are adjusted according to experimentally measured transverse cross sections. For example, for a given transverse weld profile, a unique set of input conditions may in some cases not be specified.
2. There can be a lack of information concerning the detailed characteristics of the energy source, e.g., arc energy distribution or influences of filler-metal additions on the coupling of energy into the workpiece, to name a few.
3. The system consisting of workpiece and filler metal may not have a simple geometric shape. The top surface of the melt pool may assume a relatively complex shape which may not be modelled conveniently via mathematical representations which are analytical solutions to the heat-conduction equation defined within domains having shapes characterized by relatively simple geometries.
4. The material properties are not constant and can vary substantially as a function of temperature.
5. Energy transfer within the weld melt pool is coupled to fluid convection and changes in density.

6. Energy and filler metal deposition can be such that the resulting transverse cross section of the weld is asymmetric. This situation can occur, for example, in multipass welding processes where energy and filler metal deposition can occur within an irregularly shaped groove.

A general framework for the analysis of welds using constraints, which is adaptive for the inclusion of *a priori* information (that is either obtained from experimental measurements or based on physical assumptions concerning the welding process), follows from a finite weighted-sum representation of Eq. (6.1), i.e.,

$$T_p^{(n+1)} = \frac{1}{W_p^{(n)}}\left[\sum_{i=1}^{6} k_i^{(n)} T_i^{(n)} + w_p^{(n)} T_p^{(n)}\right] \tag{6.4}$$

$$W_p^{(n)} = \sum_{i=1}^{6} k_i^{(n)} + w_p^{(n)} \tag{6.5}$$

$$w_p^{(n)} = \frac{\rho(T_p^{(n)}) C_p(T_p^{(n)})(\Delta 1)^2}{\Delta t} \tag{6.6}$$

where $k^{(n)} = k\ T^{(n)}$, the subscripts p or i designate the node and the superscripts designate the iteration with which a given quantity is to be associated. In the development that follows we define the discrete quantities:

$$\kappa_p^{(n)} = \frac{1}{6}\sum_{i=1}^{6} \kappa_i^{(n)} \quad \text{and} \quad \kappa_i^{(i)} = \frac{k_i^{(n)}}{\rho(T_p^{(n)}) C_p(T_p^{(n)})}. \tag{6.7}$$

It is important to note that in the weighted sum Eq. (6.4) the quantities $k_i^{(n)}$ represent values of the conductivity at separations $\Delta l/2$ relative to node p, whereas the quantities $T_i^{(n)}$ represent values of the temperature at separations Δl relative to node p. The specific discretization adopted for constructing the finite weighted-sum given by Eq. (6.4) represents a special case of that employed by the SIMPLE algorithm[9]. It is to be noted, however, that within our present development the sum formulation defined by Eq. (6.4) assumes a different interpretation than that associated with application of the SIMPLE algorithm. Referring to Eq. (6.4), we note that the discrete numerical equivalent of the method of distributed heat sources can be achieved by adjusting the values of the terms at the upstream or input boundary plane. This procedure tends to compensate for any lack of information concerning the detailed characteristics of the energy source and influences of filler-metal additions on the coupling of energy into the workpiece if there is sufficient information characterizing dynamic weld structure downstream. Further, this procedure is well-posed for modelling melt pool structures that are asymmetric. In the case of multipass welding it is typical that the region defined by previous passes (i.e., the groove), within which the filler metal is deposited is not symmetric. As a result, the deposition of energy is also not symmetric. Adjustment of the calculated temperature field according to geometric constraints provides a means for including information concerning asymmetry with respect to energy and filler-metal deposition. It should be noted that the asymmetry associated with the steady-state transverse cross section can be adopted as a guide for initial estimates and subsequent adjustments of the upstream boundary of the system. In what follows we show that, in the application of constraints, the pseudo steady state of the weld can be represented alternatively by either of two types of equations, where each is characterized by specific uniqueness properties. One is that of a parabolic equation where the pseudo steady state is represented by a solution of the heat conduction equation defined over an open domain. The other is

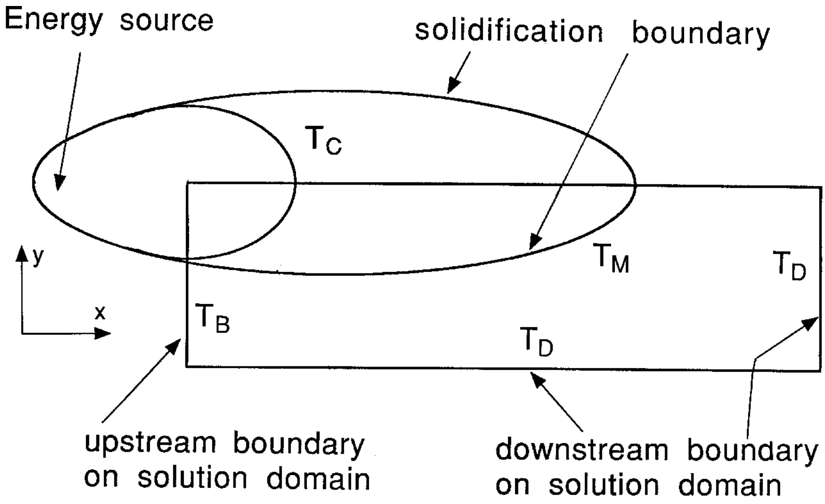

Energy source
solidification boundary
T_C
T_M
T_D
y
x
T_B
T_D
upstream boundary on solution domain
downstream boundary on solution domain

FIGURE 6.1 Schematic showing different parts of pseudo steady-state of weld and their significance with respect to the calculation of temperature fields using geometric constraints.

that of an elliptic equation where the state of the weld is represented by the Laplace equation (or Poisson equation in general) over a closed domain.

Given a representation of the pseudo-steady-state temperature field via a parabolic equation, specification of the two-dimension temperature distribution on the upstream boundary and top surface boundary (over the downstream region) is equivalent to specification of the three-dimensional distribution for deposition of energy into the material resulting from an energy source located upstream (above the upstream boundary). This equivalence is described schematically in Fig. 6.1. Referring to Fig. 6.1, it follows from the conservation of energy that the heat input per unit distance along the direction of relative motion between the energy source and workpiece provides a global constraint on the three-dimension temperature distribution extending over the downstream portion of the pseudo steady state of the weld. It is to be noted, however, that this constraint cannot be adopted as a strong condition on the temperature field in cases where the efficiency for the coupling of energy into the workpiece is not known with sufficient accuracy. In any case, however, this constraint does provide a bound condition on the system. The significance of this bound condition in terms of the theory of constrained optimization is discussed further below.

The weighted finite-sum defined by Eq. (6.4) provides a means for adopting a more general approach for inclusion of information concerning a given weld process. We have designated this approach the method of geometric constraints.[1,2] This method follows from an extension of the definition of the quantities $T_i^{(n)}$ and $T_p^{(n)}$ (defined in Eq. (6.4)) and from the mathematical properties of a closed and bounded solution domain over which the pseudo steady state temperature field of a weld is represented by an elliptic equation. Accordingly, by adopting an interpretation based on constrained optimization theory, the quantities $T_i^{(n)}$ and $T_p^{(n)}$ defined in Eq. (6.4) are assumed to be contained in a set of quantities such that each provides for the input of a specific type of information. That is to say,

$$T_i^{(n)} \quad \text{and} \quad T_p^{(n)} \in \{T, T_B, T_C, T_E, T + \Delta T_S\} \tag{6.8}$$

where T is represents those discrete temperature-field quantities determined according to the procedure defined by Eq. (6.4) whose values vary with each iteration until they are optimized subject to the constrained or specified values of the discrete temperature-field quantities T_B, T_C, T_E, and ΔT_S. The quantity T_B

represents boundary values corresponding to the temperature field on the upstream boundary. Formally, this implies that

$$T_i^{(n)} \quad \text{or} \quad T_p^{(n)} = T_B. \tag{6.9}$$

It should be noted that in the present development we consider explicitly two different types of upstream values of the temperature field and that only one of these types is associated with Eq. (6.9). One type represents upstream values of the temperature field which are boundary values of the solution domain. The other type represents upstream values of the temperature field at locations above the upstream boundary and outside the solution domain. That is to say, upstream values of the temperature field corresponding to a distribution of source terms used to construct the upstream boundary of the solution domain. The quantity T_C represents constrained values of the temperature field corresponding to information which is known *a priori* about the pseudo-steady-state temperature field associated with a given welding process. Formally, this implies that

$$T_i^{(n)} \quad \text{and} \quad T_p^{(n)} = T_C \tag{6.10}$$

for specified integer values of the node indices i and p. The quantity T_E represents values of the pseudo-steady-state temperature field at locations within the workpiece of weld and provides a means for extending the range boundary values beyond that of the upstream boundary. That is, T_E represents values of the temperature field on either the downstream boundary of the entire solution domain or boundary values on part of the surface of a closed subdomain contained within the solution domain. With respect to a given subdomain, the quantity T_E can represent either upstream or downstream boundary values. In addition, for welding processes resulting in welds which are not symmetric with respect to a longitudinal midplane, the quantity T_E can in principle represent boundary values either on the top surface boundary or midplane boundary. A partitioning of the solution domain into subdomains, upon whose boundaries constraint information can be specified, follows from the strong uniqueness condition discussed below. This condition establishes that the pseudo-steady-state temperature field of a welding process, which is represented by the heat conduction equation (parabolic-equation representation) with a given source term, can be reformulated such that it is represented by the Laplace equation (elliptic representation) within a closed and bounded domain.

In practice, the discrete quantities T_B and T_E can be assigned conveniently by means of a generating function consisting of a combination of analytical solutions to Eq. (6.1). That is to say,

$$T_B(x) \quad \text{or} \quad T_E(x) = \sum_{m=1}^{N} w_m T_m(x - x_m, \kappa_m) \tag{6.11}$$

where each function $T_m(x,\kappa)$ is an analytical solution to Eq. (6.1) corresponding to a given source term and workpiece geometry and where x_m ($m = 1,\ldots, N$) are the positions of the energy-source terms which are located upstream and outside the solution domain. The set of functions $T_m(x,\kappa)$ typically would include solutions to Eq. (6.1) of the Rosenthal type, i.e., of analytical solutions to Eq. (6.1) by Rosenthal[10,11] or of modifications of these solutions. With respect to an elliptic equation representation, the entire boundary of the solution domain, i.e., both upstream and downstream boundaries, provides a means for the inclusion of information into a model system, rather than only the upstream boundary. This follows since the upstream boundary, considered separately, is an open boundary which represents only a partial section of the total boundary of the solution domain (see discussion in Reference 1). It can be shown, however, that there exists a range of different distributions of localized energy-source terms that can generate two-dimensional cross sections defined by a given two-dimensional isothermal boundary. In what follows we present a series of calculations which demonstrate this property. In contrast to the distribution of values defining only the upstream boundary of the solution domain, the distribution of values defining the entire closed boundary of the solution domain is a global quantity that is subject to relatively stronger uniqueness criteria. This issue is discussed in some detail in Reference 2. The quantity ΔT_s represents increments to the field values resulting from source terms located within the system which are represented explicitly. Specification of the quantity ΔT_s can facilitate the inclusion of information concerning the energy distribution characteristics of the energy source.

6.4 Theoretical Foundation of the Geometric-Constraints Method

The geometric constraints method which we have adopted represents a reformulation of the problem of weld analysis such that it can be stated in terms of a system-parameter optimization problem.[3,4,5] In terms of this reformulation Eq. (6.4) is not to be interpreted as a discrete finite-difference approximation of Eq. (6.1) which is to be determined iteratively according to a given set of boundary conditions. In terms of the geometric constraints method, Eq. (6.4) is interpreted as a weighted finite-sum defined for the purpose of constructing a quantity which is designated, in the theory of constrained optimization, as the *objective function*. A formal statement of the classical constrained optimization problem which provides the basis of our method is as follows.

The optimum set of values for a given set of parameters characterizing a system is that corresponding to the minimum of the objective function

$$Z = f(X), \tag{6.12}$$

where $X = (X_1, X_2, \ldots, X_N)$ are the set of parameters characterizing the system, subject to the constraints

$$g_i(X) = b_i, \tag{6.13}$$

where the functions $f(X)$ and $g_i\,(X)$ (i = $1, \ldots, m$) have continuous first derivatives.

The strict interpretation of Eq. (6.4) as a weighted finite-sum, rather than as a finite-difference approximation, is adopted in order to be consistent with the general approach of adopting as system parameters a finite set of discrete field quantities. In our present development this set consists of discrete field quantities representing temperature, conductivity and diffusivity, where each quantity is associated with a spatial location indexed by an integer variable p.

An example of an objective function whose minimum would correspond to the constrained optimum of a discrete temperature field T_p ($p = 1, \ldots, N_S$) relative to a given set of geometric constraints is described with reference to Fig. 6.1. This formulation of an objective function is based on a separation of the temperature field of the pseudo steady state of the weld into three regions. These regions are: (1) the solution domain, the region bounded by the upstream and downstream boundaries; (2) the region of the weld which is above the upstream boundary of the solution domain; and (3) the region of the weld which is below the downstream boundary of the solution domain (and whose associated range of temperatures is above the ambient temperature of the workpiece). Each of these regions is partitioned into a set of discrete volume elements of volume (Δl).[3] Given this partitioning, an objective function for constrained optimization of the discrete field T_p, whose general form is meant to be representative in the present development, is given by the absolute value of

$$\begin{aligned} Z_{BD} = {} & \frac{q_0}{V_B} - \frac{(\Delta l)^3}{L_S}\sum_{p=1}^{N_S}\int_{T_A}^{T_P}\rho(T)C_p(T)dT \\ & -\frac{(\Delta l)^3}{L_B}\sum_{p=1}^{N_B}\int_{T_A}^{T_P}\rho(T)C_p(T)dT - \frac{(\Delta l)^3}{L_D}\sum_{p=1}^{N_D}\int_{T_A}^{T_P}\rho(T)C_p(T)dT \end{aligned} \tag{6.14}$$

where

$$\frac{(\Delta l)^3}{L_S}\sum_{p=1}^{N_s}\int_{T_A}^{T_p}\rho(T)C_p(T)dT \rightarrow \frac{q_0}{V_B} \tag{6.15}$$

as N_B, N_D and Δl decrease and N_S increases and where the quantities q_0 and V_B are the power input and weld speed, respectively. The significance of taking the absolute value of Z_{BD} to construct an objective function is discussed below. Although the choice of the specific form of Eq. (6.14) is arbitrary in our present development, there are general features of this equation which have general significance. The second term of this expression represents a sum over discrete field quantities that are within a closed solution domain whose length is L_S. In principle, this region can be extended to include the entire temperature field associated with the pseudo steady state of the weld, including the region containing the energy source and all values of the discrete temperature T_p that are above the ambient temperature of the workpiece. For this case the remaining terms are essentially zero. The third term in Eq. (6.14), representing a sum over discrete field quantities which extend over a domain whose length is L_B, is taken to represent explicit and detailed information concerning the characteristics of the energy source. The last term in Eq. (6.14), extending over a domain whose length is L_D, is taken to represent the temperature field of the pseudo steady state that is not within the specified solution domain.

Another set of expressions upon which the formulation of an objective function can be based follows from the weighted finite-sum defined by Eq. (6.4). Because the model system is that of a pseudo-steady state rather than that of a steady state, the discrete quantities Δt and Δl are not independent and are related by

$$\Delta t = \frac{\Delta l}{V_B}. \tag{6.16}$$

It follows that, if the solution domain is partitioned into a set of discrete volume elements of volume $(\Delta l)^3$, there can be associated with each of these elements an objective function given by the absolute value of

$$Z_p = \sum_{i=1}^{6}\kappa_i T_i - 6\kappa_p T_p - (V_B\Delta l)(T_p - T_{p+1}), \tag{6.17}$$

where the node $(p + 1)$ is located both adjacent and upstream relative to node (p). Equation (6.17) is equivalent to the combination of Eq. (6.4) and the condition defined by Eq. (6.16). An example of an objective function whose minimum would correspond to the constrained optimum of a discrete field T_p which adopts (in terms of constraints) information concerning both the energy input and shape features of a weld, e.g., experimental weld cross sections and thermocouple measurements, is given by

$$Z = \left|Z_{BD} + \sum_{p=1}^{N_S} Z_p\right|, \tag{6.18}$$

where the quantities Z_{BD} and Z_p are defined by Eq. (6.14) and Eq. (6.17), respectively, and N_S is the number of nodes making up the solution domain and depends on the location of the downstream boundary. The location of the downstream boundary is selected according to the type of constraint information that is available. The significance of the absolute value in Eq. (6.18) follows from optimization theory and concerns general criteria for establishing a well posed optimization problem. An optimization procedure will tend to be well posed if the objective function to be minimized is convex, i.e., among the

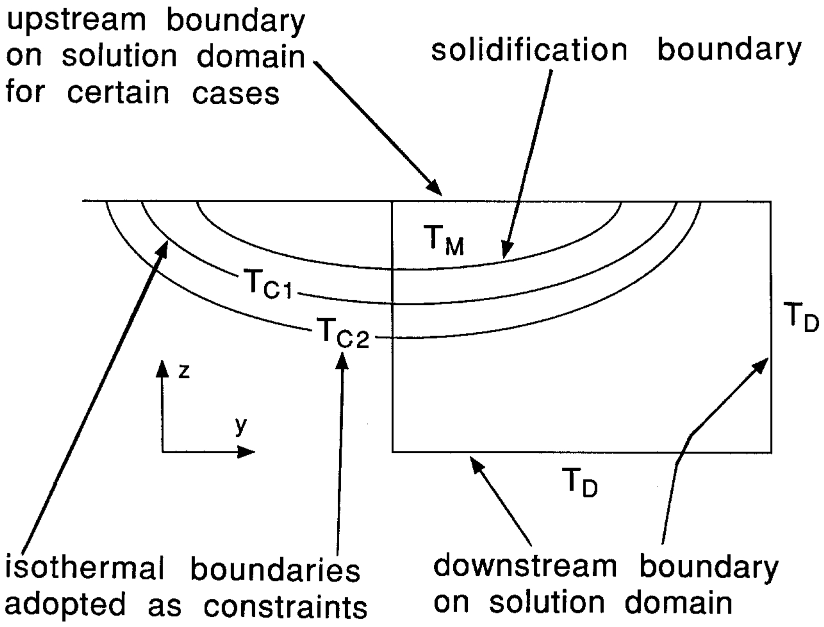

upstream boundary on solution domain for certain cases
solidification boundary
T_M
T_{C1}
T_{C2}
T_D
z
y
T_D
isothermal boundaries adopted as constraints
downstream boundary on solution domain

FIGURE 6.2 Schematic showing different parts of transverse cross section of weld and their significance with respect to the calculation of temperature fields using geometric constraints

special class of functions characterized by a global minimum within their domain of definition. Accordingly, the objective function defined by Eq. (6.18) is in principle such that $Z \geq 0$ and has a global minimum at $Z = 0$. A brief summary of properties of convex functions that are relevant to constructing an objective function is given in Appendix C.

In general, the geometric constraint conditions are of three types. These are, referring to Fig. 6.2.

$$\max\{T(x_i, y_M, z_M)\} = T_M, \tag{6.19}$$

$$\max\{T(x_i, y_C, z_C)\} = T_C, \tag{6.20}$$

$$T(x_M, y_M, z_M) = T_M, \tag{6.21}$$

and

$$T(x_C, y_C, z_C) = T_C. \tag{6.22}$$

Adopting the formal terminology of optimization theory, we refer to the constraints given by Eq. (6.19) and Eq. (6.20) as passive constraints and those given by Eq. (6.21) and Eq. (6.22) as active constraints. Equations (6.19) and (6.21) represent passive and active constraint information, respectively, concerning the solidification boundary. It can be noted that the method of distributed heat sources[3] corresponds to the special case of minimizing the objective function Eq. (6.18) subject to the type of constraints defined by Eq. (6.19) for temperature independent material properties. The immediate extension of this method follows by adopting the active constraint conditions defined by Eq. (6.21). Analysis of deep penetration weld structure based on this immediate extension is the subject of References 1 and 2. The general method of geometric constraints follows by adopting the conditions defined by Eq. (6.20) through Eq. (6.22) which represent passive and active constraint information concerning isothermal surfaces and specified temperature values within the solution domain. This information could be associated with, for example, surfaces corresponding to the onset or final stages of phase transitions, e.g., Ac_1 and Ac_3, or thermocouple measurements. Reed and Bhadeshia[12] have adopted information concerning Ac_1 and Ac_2 for calibration of their model for multipass steel welds. This model is based on a Rosenthal type solution. The addition of constraint conditions based on information concerning the shape features of an energy source poses the possibility of constructing additional terms for an objective function which would entail the quantities ΔT_S defined in Eq. (6.8).

6.5 Some Properties Associated with Geometric Constraints

In this section we discuss a series of properties which either follow from the general formulation of the geometric constraints method or are relevant to its application in practice for the analysis of weld morphology.

Quasi-one-dimensionality of weighted finite-sum representation (an approximate condition for uniqueness)—The quasi-one-dimensional character of the pseudo-steady state provides a basis for establishing an approximate condition for the uniqueness of the discrete temperature field T_p. This condition follows from uniqueness properties of ordinary differential equations.The pseudo-steady state imposes the condition that the change in temperature ΔT_p at a given node p, located downstream relative to node $p + 1$, is given by

$$\Delta T_p = T_p - T_{p+1} \tag{6.23}$$

for a period of time Δt defined by Eq. (6.16). Next, adopting a node indexing scheme where nodes $p - 1$, p and $p + 1$ are along an axis parallel to the direction of relative motion of the energy source, Eq. (6.4) can be expressed in the form

$$\kappa_{p-1} T_{p-1}^{(n)} + S_p + (\kappa_{p+1} + V_B \Delta l) T_{p+1}^{(n)} = 0 \tag{6.24}$$

where

$$S_p = \sum_{i=1}^{4} \kappa_i T_i^{(n)} - [6\kappa_p + (V_B \Delta l)] T_p^{(n+1)}. \tag{6.25}$$

It is seen that relative to an axis that is parallel to the direction of beam travel, Eq. (6.24) assumes the form of a quasi-one-dimensional second-order difference equation. This property is significant in that it provides an approximate condition for assuring uniqueness of the calculated temperature field for a given set of geometric constraints and upstream boundary conditions on the solution domain. Minimization of an objective function such as that defined by Eq. (6.18) by means of an iterative procedure defined by Eq. (6.4) is such that the uniqueness of the converged solution follows from closure conditions on the solution domain. That is to say, one knows *a priori* a sufficient amount of information concerning the temperature field both at distributed positions within and over the entire boundary of the solution domain to adopt constraints which bound and distributively weight the solution sufficiently to assure that the minimum of the objective is essentially unique. On the other hand, minimization of an objective function, e.g., Eq. (6.18), by means of a procedure defined by Eq. (6.24), is essentially a marching process where the uniqueness of the solution follows from conditions only at the upstream boundary of the solution domain. For this case, where generation of the solution is strictly due to upstream conditions, the superscripts in Eq. (6.23) and Eq. (6.24), which index the iterations, still have significance in terms of lateral weighted of the temperature field with respect to the direction of beam travel. This follows since the temperature field is not strictly one-dimensional. Note that the approximate condition for uniqueness which we have described here is with respect to a representation of the pseudo steady state via a parabolic equation.

Duality between T and κ—An important aspect of the weighted finite-sum representation, e.g., Eq. (6.17), is that the discrete field quantities T_i and κ_i have an equivalent status formally. That is to say, minimization of the objective function can be effected with respect to the κ field as well as the T field. This property provides the basis for further generalization of the method of geometric constraints. First, there can be cases where information concerning the thermal diffusivity $\kappa(T)$ may only be known partially. In these cases, for those regions of the solution domain where information concerning $\kappa(T)$ is either unavailable or approximate, the discrete field quantities κ_i can also be adopted as optimization parameters. The requirement is that there exists a sufficient amount of geometric constraint information for assuring a unique minimum of the objective function with respect to both T_i, spanning the entire solution domain, and the set of quantities κ_i whose values are not known *a priori*. Second, there can be cases where the available geometric constraint information may be such that the problem is over constrained. That is to say, the number of equality constraints of the type Eq. (6.21) and Eq. (6.22) exceeds the number of discrete field quantities whose values are to be optimized. In these cases, an increase in the number of discrete field quantities can be effected by adopting the quantities κ_i as well as T_i for optimization. It follows then that the duality between T and κ with respect to the weighted finite-sum representation provides a means for increasing the amount of information which can be put into a model system via constraints.

Extending the range of system parameters—An objective function such as Eq. (6.18) can be easily adapted to include additional information concerning a weld which may be associated with regions that are outside a specified solution domain. This would include detailed information concerning the heat source and would in principle represent constraints on the upstream boundary values of the solution domain. In this context, it is significant to compare minimization of the objective function defined by Eq. (6.14) to the method of distributed heat sources. Stated formally in terms of a parameter optimization problem, the method of distributed heat sources represents minimization of the objective function

$$Z = \left| q_o - \sum_{i=1}^{N} q_i \right| \tag{6.26}$$

subject to the type of constraint that is defined by Eq. (6.19) and the assumption that the value of q_o is known. The quantities q_o and q_i ($i = 1, \ldots, N$) are the total power input and individual power inputs associated with a finite distribution of sources, respectively. It follows that inclusion of any information concerning details of the energy source, i.e., its power and distribution, can be accomplished by adopting constraints on the distribution of temperatures above and including the upstream boundary of the solution domain. It is to be noted that optimization of the objective function Eq. (6.18) with the condition $Z_{BD} = 0$ is equivalent to the iterative solution of Eq. (6.4) for a given set of upstream boundary conditions. For the prototype analyses presented below, a restricted form of Eq. (6.18) is adopted which does not adopt in terms of constraints detailed characteristics of the energy source.

A reformulation of the problem (a strong condition for uniqueness)—Adopting the conditions for a pseudo steady state, i.e., Eq. (6.16) and Eq. (6.24), the weighted finite-sum Eq. (6.4) can be expressed in the form

$$T_P = \frac{1}{(6\kappa_P + V_B \Delta l)} \left[\sum_{i=1}^{6} \kappa_i T_i + (V_B \Delta l) T_{p+1} \right]. \tag{6.27}$$

Note that Eq. (6.27) can also be deduced from a Rosenthal-type solution of Eq. (6.1) (see Appendix B). Next, we adopt the substitution

$$\kappa_i = \kappa_i^*, \tag{6.28}$$

where the discrete quantities κ_i^* represent values of diffusivity which are functions of discrete positions indexed by (i) at separations of Δl relative to the position whose discrete index is (p), e.g., $\kappa_i^* = (\kappa_p + \kappa_i)/2$. Defining

$$w_i^* = \kappa_i^* + \delta_{i,p+1}(V_B \Delta l) \quad \text{and} \quad w_p^* = \frac{1}{6}\sum_{i=1}^{6} w_i^*, \tag{6.29}$$

it follows that both sets of discrete field quantities w_i^* and T_i are functions of positions which are at separations Δl relative to the position indexed by (p) and that Eq. (6.27) can be expressed as

$$T_p = \frac{1}{6w_p^*}\sum_{i=1}^{6} w_i^* T_i. \tag{6.30}$$

Equation (6.30) represents a discrete analog of the Laplace equation

$$\nabla^2 \left[w^*(T, x, V_B)T \right] = 0. \tag{6.31}$$

It is seen that within this representation of the pseudo steady state the weighted function $w^*(T, x)$ enters formally as a generalized diffusivity. There is then a unique solution to Eq. (6.31) if values of the field $w^*(T, x)T$ are specified over the entire boundary of a given solution domain in which T is to be determined. This implies that the boundary generating functions $T_B(x)$ and $T_E(x)$ are functions of $w^*(T, x)$ as well as the power input. Or, more generally, that the boundary conditions must include information concerning the relative motion of the energy source and the temperature dependence of the diffusivity. Note that the strong condition for uniqueness described here is with respect to a representation of the pseudo steady state via an elliptic equation.

Sensitivity of calculated temperature field—Sensitivity represents another type of influence on temperature fields determined by inversion of two-dimensional weld cross-section information and is in general of three types. These types are: sensitivity with respect to the total amount of energy that is input; sensitivity with respect to the spatial character of energy deposition into the weld and sensitivity with respect to the material properties of the workpiece and filler metal. An example of the influence of these types of sensitivities is given by Fig. 6.3 through 6.5. The temperature fields shown is these figures are obtained by approximate inversion of weld cross section data shown in Fig. 6.7. The parameters for this weld are given in the appendix. In each case inversion of the weld cross section is accomplished by either the method of distributed heat sources (for those cases where material properties are assumed independent of temperature) or numerically by a specification of boundary values on the upstream boundary of a bounded solution domain (for temperature dependent material properties (6.13)).

Shown in Fig. 6.3 are calculated transverse cross sections showing the Ac_1 boundary (defined by 675°C), Ac_3 boundary (defined by 840°C) and solidification boundary (defined by T_M). These cross sections correspond to relatively dissimilar welds involving different material properties for the workpiece and different powers and spatial configurations for the energy source. These results, which are consistent with the uniqueness criteria discussed above, demonstrate that very similar transverse weld cross sections (defined by a given two-dimensional isothermal boundary) can correspond to significantly different welds. Longitudinal slice sections (within L_S) of the three-dimensional temperature fields (i.e., top surface and midplane of weld) corresponding to these transverse cross sections are shown in Fig. 6.4. It is seen that these temperature fields are dissimilar and yet map into cross sections of the solidification boundary which are very similar. These results suggest that the transverse cross section can be insensitive to relatively wide variations in process parameters and other general features of a weld. Note in particular that two of the welds shown in Fig. 6.4 are for identical workpieces but involve different input powers and spatial configurations of the energy source. An example of the type of sensitivity that can occur with respect to variations in material properties is given by Fig. 6.5. The cross sections shown in Fig. 6.5 are for calculated

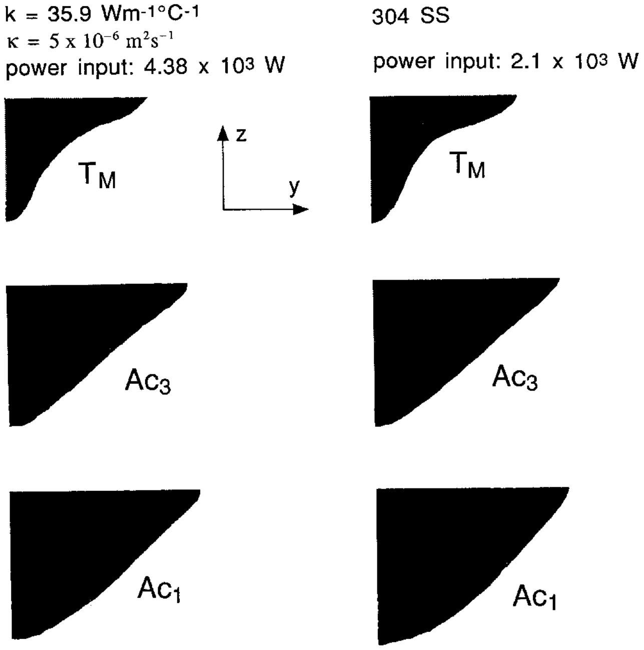


FIGURE 6.3 Prototype transverse cross sections which show a high level of similarity but correspond to significantly different welds.

temperature fields corresponding to identical energy sources and process parameters but for different material properties (which are assumed independent of temperature). Referring to Fig. 6.5, it can be seen that the calculated transverse cross sections are relatively insensitive to reasonably large variations in material properties. The insensitivity of calculated transverse cross sections to various weld process parameters implies that it is important to adopt as constraints on the calculated temperature field some type of information concerning the three-dimensional character of the weld. This information need not represent detailed knowledge of the temperature field but rather bounds on its distribution of values in three dimensions. Such information could be obtained by thermocouple measurements or from weld cross sections other than those that are tranverse.[1,2]

6.6 Prototype Case Study Analyses of Welds

We present a set of case studies which are typical of a range of different welding processes, e.g., bead-in-groove and bead-on-plate welding processes. This collection of prototype analyses serves to demonstrate many of the properties of the geometric constraints method and issues associated with its application in practice. Our analyses are based on the minimization of the objective function

$$Z = \left| \frac{q_0}{V_B} - \frac{(\Delta l)^3}{L_S} \sum_{p=1}^{N_s} \int_{T_A}^{T_p} \rho(T) C_p(T) dT + \sum_{p=1}^{N_S} Z_p \right|, \qquad (6.32)$$

where Z_p is given by Eq. (6.17). Equation (6.32) is defined with reference to Fig. 6.6 and represents a restricted form of Eq. (6.26). Some of the weld data considered in our analyses are given in data sets 1 and 2 in the appendix and described in Figs. 6.7 and 6.8. Included also in these data sets are model parameters associated with our analyses. Shown in Fig. 6.7, is a transverse cross-section for a bead-on-plate weld having some reinforcement extending above the surface of the workpiece. The process

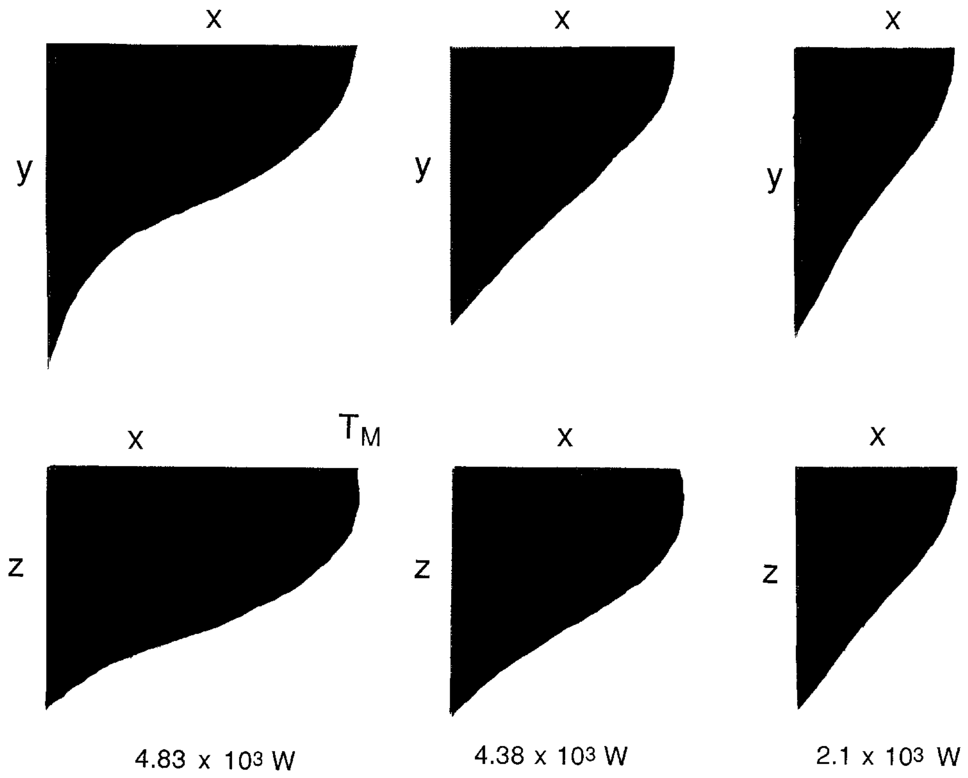


FIGURE 6.4 Comparison of temperature fields corresponding to prototype cross sections which show a high level of similarity.

parameters for this weld are given in data set 1 of Appendix A. This set of data includes information concerning projections onto the transverse cross section of the weld of the solidification boundary and of the Ac_1 and Ac_3 isotherms. Shown in Fig. 6.8 are solidification boundaries at the top surface and transverse cross sections for a bead-in-groove weld having essentially a flat surface. The process parameters for this weld are given in data set 2 of Appendix A. This set of data includes, in addition to information concerning the shape of the solidification boundary, information concerning the projection of the Ac_1 isotherm onto the transverse cross section.

Prototype case study 1. General procedure for constrained optimization of discrete temperature field—For the present case study we consider a prototype analysis of the weld whose cross section is shown in Fig. 6.7 and whose process parameters are given in data set 1 of Appendix A. For this analysis only the transverse cross section of the solidification boundary is adopted as a constraint based on experimental measurements. We adopt as an objective function the expression defined by Eq. (6.32). Minimization of the value of this function represents an optimization of the energy per distance for a finite region of the weld spanning the calculated temperature field which does not totally include the energy source. This region does, however, include a large fraction of the weld, i.e., the region defined by L_S in Fig. 6.6. For this analysis we adopt a three-dimensional solution domain defined by six boundaries which enclose a finite section of the total region spanning the pseudo steady state. Figure 6.1 provides a two-dimensional schematic representation of the top view of this model system. The upstream boundary is considered to be either in close proximity or partially within the region of the weld within which there is energy deposition. One of the longitudinal boundaries corresponds to the midplane of the weld. The top surface boundary corresponds to the top surface of the workpiece and represents an approximation since for this weld (data set 1 of Appendix A) there is substantial reinforcement extending above the workpiece surface. Three of the boundaries are adopted as downstream boundaries on the solution domain.

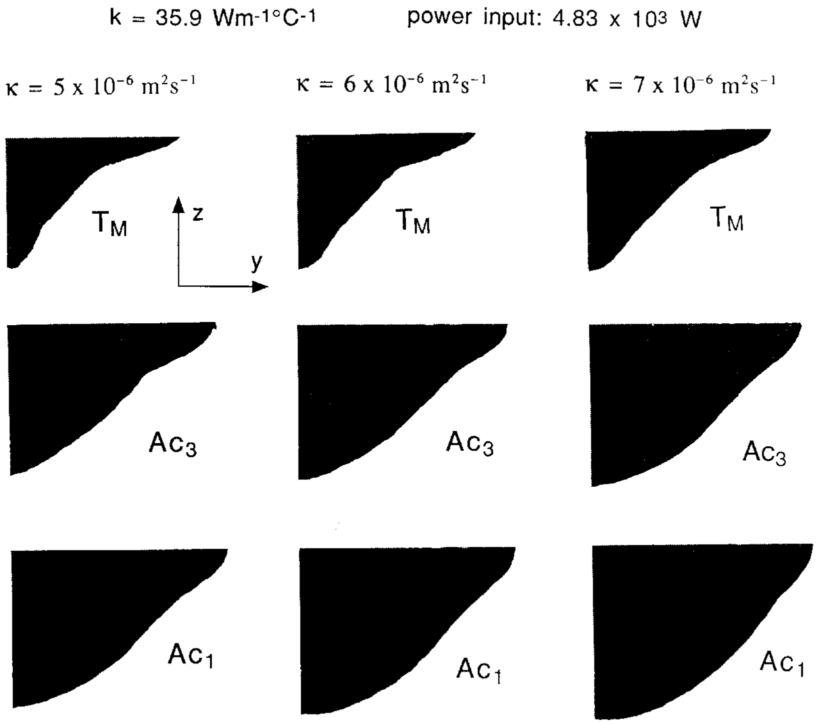


FIGURE 6.5 Comparison of prototype cross sections for identical welding conditions and workpieces having different thermal properties.

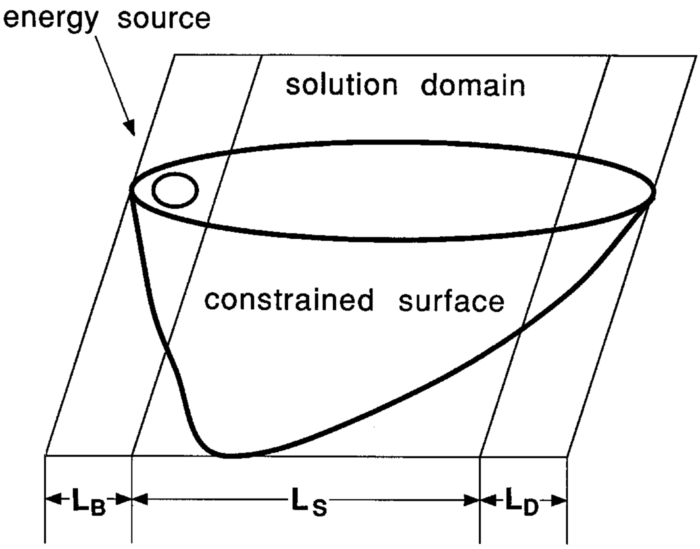


FIGURE 6.6 Schematic representation of partitioning of pseudo steady state of weld for terms defined in Eq. (6.14).

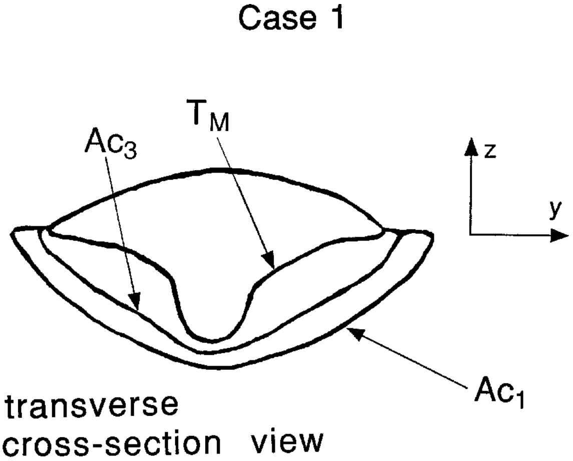


FIGURE 6.7 Experimentally measured shape of solidification boundary at top surface and transverse cross section of meltpool corresponding to welding parameters for case 1. Included is a prototype *Ac*1 boundary.

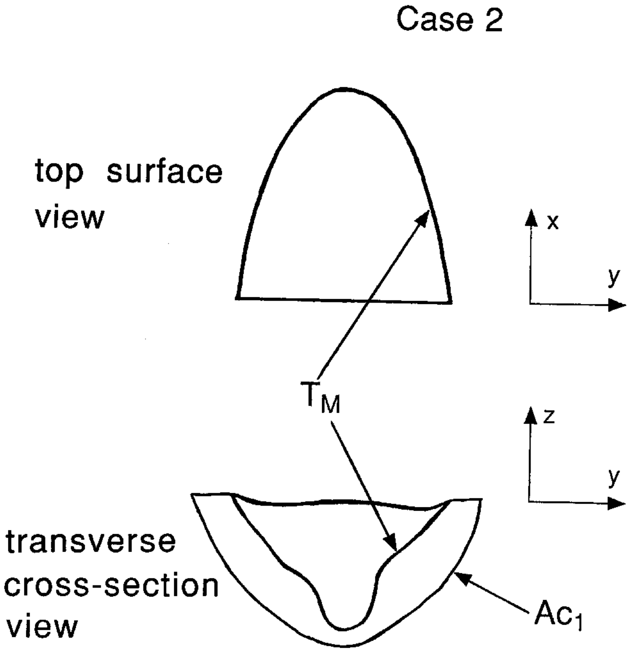


FIGURE 6.8 Experimentally measured tranverse cross section of meltpool corresponding to welding parameters for case 2. Included are prototype *Ac*1 and *Ac*3 boundaries.

For this case the first stage of our analysis entails the generation of a three-dimensional isothermal surface at T_M whose transverse projection maps onto the experimentally measured transverse cross section of the solidification boundary. This procedure is undertaken by means of a generating function $T_E(x)$ defined by Eq. (6.11). At this stage, one may assume various levels of approximation since the primary objective at this point is the generation of an isothermal surface and not of the temperature field defined over the entire solution domain. If one adopts, however, a reasonably accurate Rosenthal-type solution as a generating function $T_E(x)$ then there will be generated (in addition to an experimentally consistent solidification

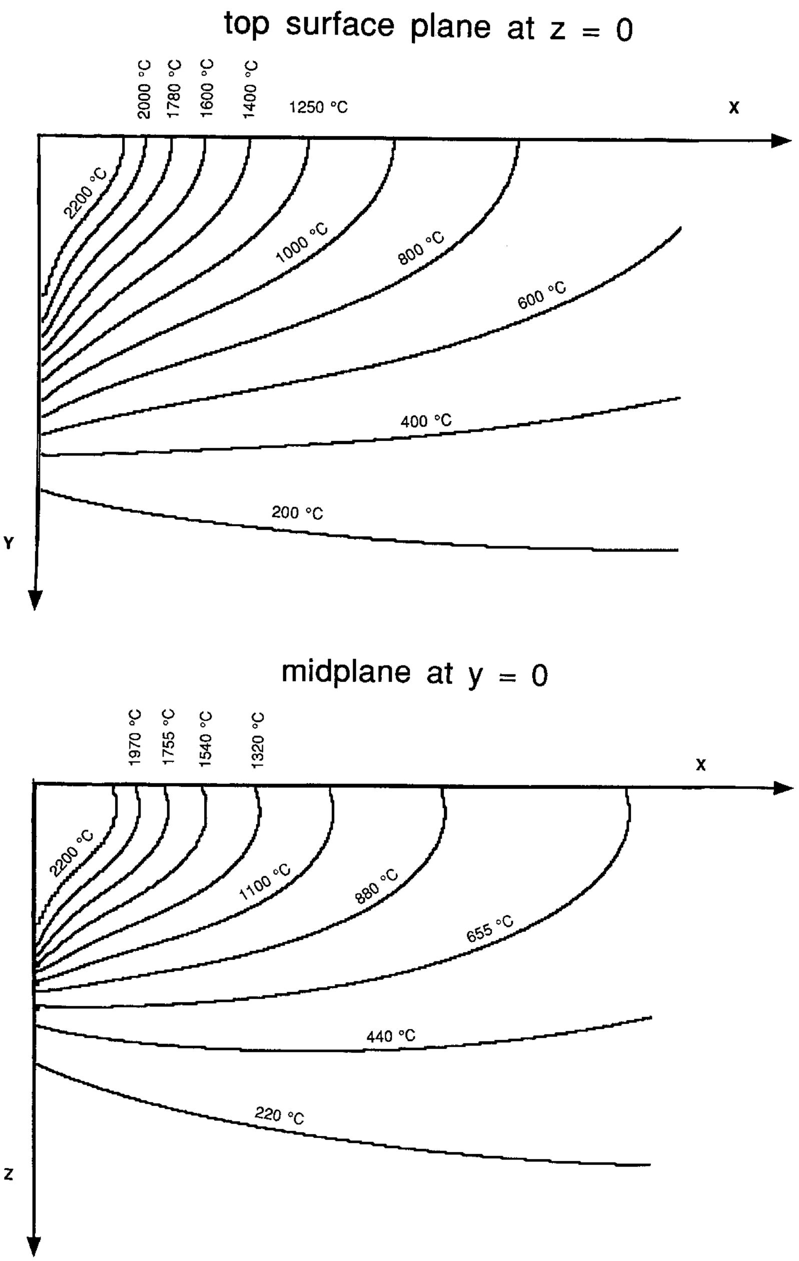


FIGURE 6.9 Cross sections of estimated temperature field calculated by means of a generating function with the condition that the isothermal surface at T_M is consistent with Fig. 6.8. The temperature fields shown in Figs. (6.9a) and (6.9b) are those on the top surface and symmetry planes, respectively, of the model system.

boundary) a reasonably close estimate of the temperature field at other regions of the solution domain. Note that for this case, however, adopting the temperature field generated by $T_E(x)$ as a very close estimate of the solution represents the equivalent of the method of distributed heat sources and does not permit inclusion of any constraints other than those associated with a single surface extending within the solution domain.

Shown in Fig. 6.9 are two-dimensional cross sections of a temperature field, calculated via a generating function $T_E(x)$, for which the isothermal surface at T_M is consistent with Fig. 6.8. There are some features

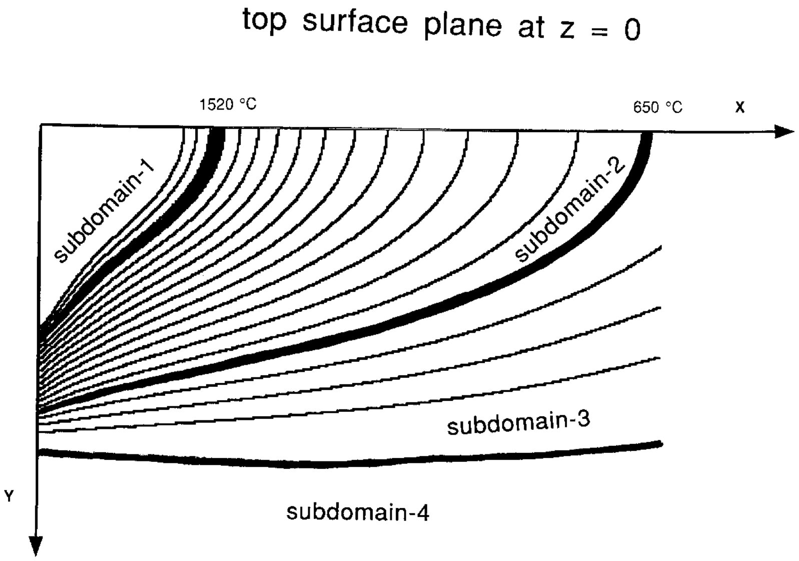


FIGURE 6.10 Cross-section of temperature field showing an example of partitioning of solution domain into subdomains for the purpose of embedding information concerning a weld.

of this particular calculation that should be noted. First, for the region of the solution domain close to the upstream boundary and greater than T_M, the calculated temperature field does not represent any type of reasonable estimate, or for that matter, any physically realistic temperature distribution. For the calculations shown in Fig. 6.9, the primary condition on the function $T_E(x)$ is that of generating an isothermal surface at T_M that is consistent with Fig. 6.7. Second, for this calculation we have adopted a generating function which is symmetric with respect to the midplane boundary and which consists of a sufficiently large number of terms in each of the Rosenthal-type solutions comprising $T_E(x)$ that a relatively high level of accuracy is maintained. In addition, we have used $T_E(x)$ to generate field values over the entire solution domain. As a result, the values generated using $T_E(x)$ should represent a realistic estimate of the temperature field for regions of the solution domain that are not close to the upstream boundary. This level of accuracy, however, is not necessary at this stage of the procedure. The level of computation associated with generating the potentially constrained surfaces can be reduced by using a generating function which is not necessarily symmetric with respect to the midplane of the weld, does not necessarily represent a fully converged Rosenthal-type solution and which is only used to generate temperature values in regions of the solution domain that are either on or close to those surfaces to be constrained. It is significant to note that any approximations or unrealistic distributions of temperature that are generated at this stage (within the solution domain) are removed by optimization of the objective function Eq. (6.32) for a given distribution of temperatures on the upstream boundary. And thirdly, the region corresponding to the solution domain does not include the entire pseudo steady state of the weld. Therefore we are not, for the present case, considering a full optimization of the objective function with respect to input energy. The relative significance of achieving a relatively full optimization of Z with respect to q_0/V_B is discussed below and in the appendix.

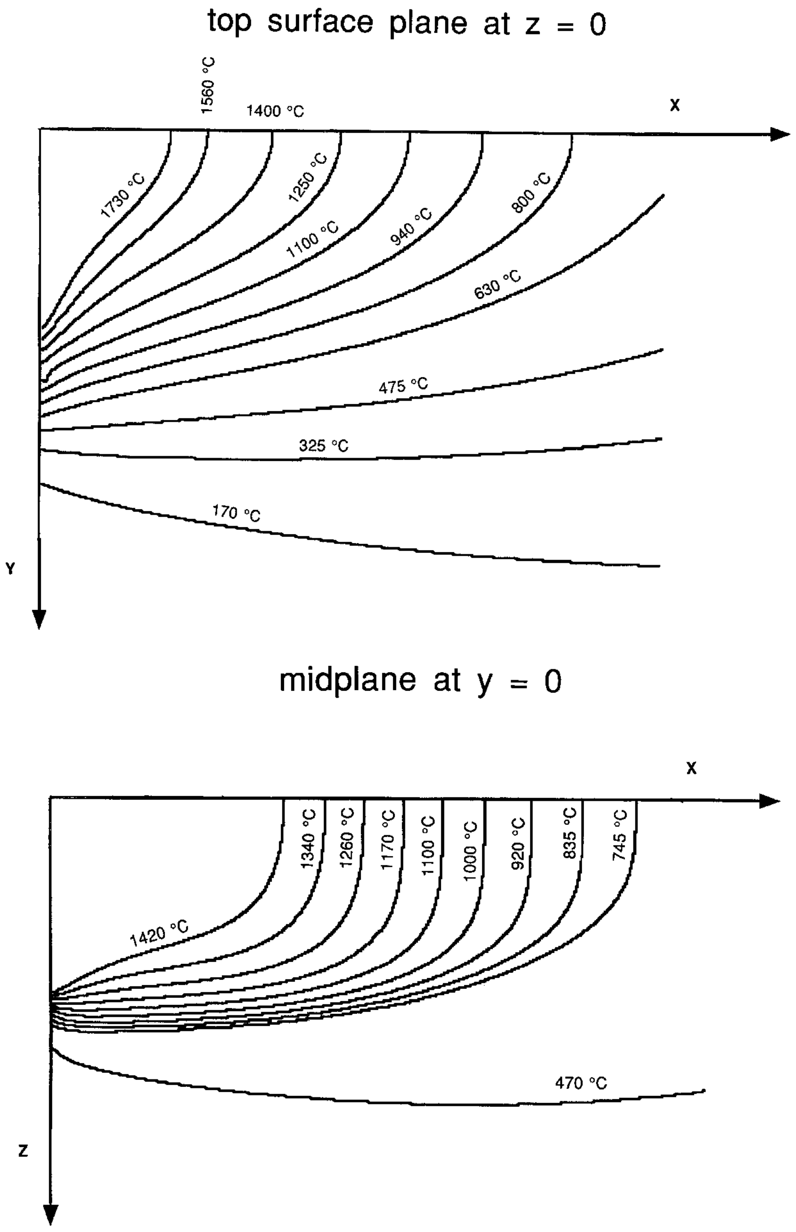
The second stage of our analysis consists of partitioning the solution domain into subdomains. It is this stage of our procedure which provides the basis for embedding known information about the weld into the solution domain. An example of this partitioning is shown in Fig. 6.10, where the solution domain is partitioned into four subregions, each defined by a given set of upstream and downstream boundaries. Subdomain 1 in Fig. 6.10 is defined by a closed boundary consisting of a section of the upstream boundary of the solution domain, a section of the midplane boundary and the isothermal surface at T_M. Subdomain 2 in Fig. 6.10 is defined by a closed boundary consisting of a section of the upstream boundary of the solution domain, the isothermal surface at T_M, a section of the midplane boundary and the isothermal surface at 650°C. Subdomain 3 in Fig. 6.10 is defined by a closed boundary consisting of a section of the upstream boundary of the solution domain, the isothermal surface at 650°C, a section of the midplane boundary, the isothermal surface at 350°C, and a section of the downstream boundary of the solution domain. Subdomain 4 in Fig. 6.10 is defined by a closed boundary consisting of a section of the upstream boundary, the isothermal surface at 350°C, and various portions of the three downstream boundaries of the solution domain.

The third stage of this analysis entails the specification of a distribution of temperatures on the upstream boundary of the solution domain. This procedure is undertaken by means of a generating function $T_B(x)$ which is defined, as for the function $T_E(x)$, by Eq. (6.11). For a specific partitioning of the solution domain into a given set of subdomains, the objective function Eq. (6.32) is then minimized over the solution domain for a given distribution of temperatures on its upstream boundary. Shown in Fig. 6.11 are two-dimensional slices of a temperature field whose discrete values minimize the function Eq. (6.32) for the partitioning of the solution domain shown in Fig. 6.10 and a given distribution values on its upstream boundary.

There are features of the second and third stages of our analysis and of this particular calculation that should be noted. First, we have restricted our minimization of Eq. (6.32) only with respect to the subdomains 1, 2, and 3 in Fig. 6.10. For this calculation the generating function $T_E(x)$ used to generate the boundary at T_M provides, in addition, a sufficiently good estimate of the temperature field at downstream regions of the pseudo-steady state (i.e., T below 350°C) that the values of this function may themselves be adopted as the solution within subregion 4 of Fig. 6.10. That is to say, for this calculation we have in fact adopted a hybrid procedure in which the method of distributed heat sources is used to calculate the temperature field for T above 350 °C and the method of constraints is used for temperatures below 350°C. Second, by partitioning the solution domain into four subdomains, we have provided, in principle, a means for the inclusion into our model system of information corresponding to experimental measurements taken at three separate regions of the dynamic weld. An example of this type of information is shown in Fig. 6.8, where it is assumed that (in addition to information concerning the solidification boundary) there is information related to the Ac_1 and Ac_3 boundaries. Another example of this type of information is that which would be obtained from thermocouple measurements. For the calculated temperature field shown in Fig. 6.11, the subdomain boundaries at temperatures other than T_M would then be considered as representing constraints corresponding to specific *a priori* information concerning the distribution of temperatures in the weld. Lastly, the partitioning of the solution domain into a relatively distributed set of subdomains represents an aspect of our procedure which has significance beyond that of providing a means for the inclusion of *a priori* information about weld temperatures. This stage of our procedure, which may appear somewhat artificial if used without the inclusion *a priori* information, actually provides a means for enhancing the accuracy of the calculated temperature field by helping to make the objective function better conditioned. The underlying mathematical properties providing the basis for this improved conditioning of the objective function are relatively subtle and are discussed below.

The final stage of this analysis entails in principle optimization of the temperature field with respect to energy input. This stage is accomplished by repetition of the third stage of our analysis (described above) according to adjustments of the distribution of upstream boundary values on the solution domain. It should be noted that without relatively detailed information concerning the energy source, a full optimization with respect to energy input is not in general well posed. That is to say, adjustment of boundary values such that the objective function Eq. (6.32) obtains its minimum value with respect to energy input may not be relevant. It is reasonable to obtain, however, a partial optimization where a discrete temperature field is sufficiently optimum that the value of the objective function is a reasonably close estimate of the minimum with respect to the energy input for a given set of constraints. It should be noted further, that a discrete temperature field which is optimized under a sufficiently large set of constraints will tend implicitly toward being optimum with respect to the energy input.

Before proceeding, we examine those features of our method that are related to the partitioning the solution domain into a distributed set of subdomains. First, a distributed partitioning of the solution


top surface plane at z = 0

midplane at y = 0

FIGURE 6.11 Cross-sections of discrete temperature field whose values have been optimized with respect to the constraint conditions shown in Fig. 6.10 and a given set of upstream boundary values.

domain helps to make a procedure entailing the constrained optimization of a bounded finite set of quantities better posed for calculating a discrete temperature field. That is, in contrast to a procedure entailing numerical integration of a discrete system of equations based on finite-difference approximations. As discussed previously, the weld-analysis problem is such that Eq. (6.1) is stiff because of characteristically large changes with respect to position of the temperature gradient for positions increasingly close to the energy source.[14] It follows then that an objective function consisting of a finite weighted sum which is weighted mostly by upstream values will tend to be poorly conditioned. Minimization of the objective function with respect to quantities within a bounded and closed subdomain is such that there

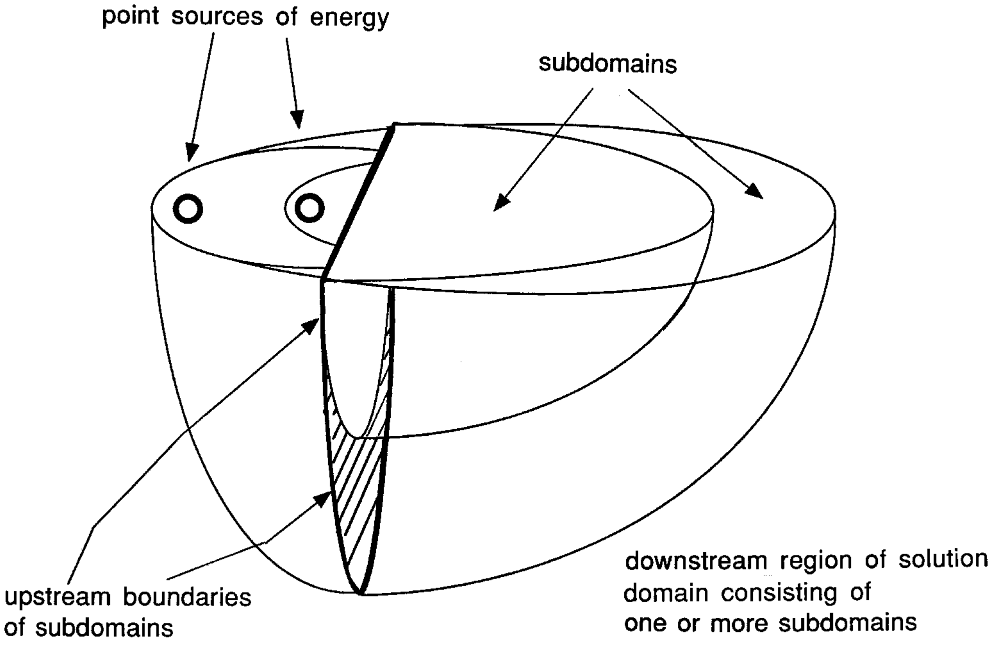
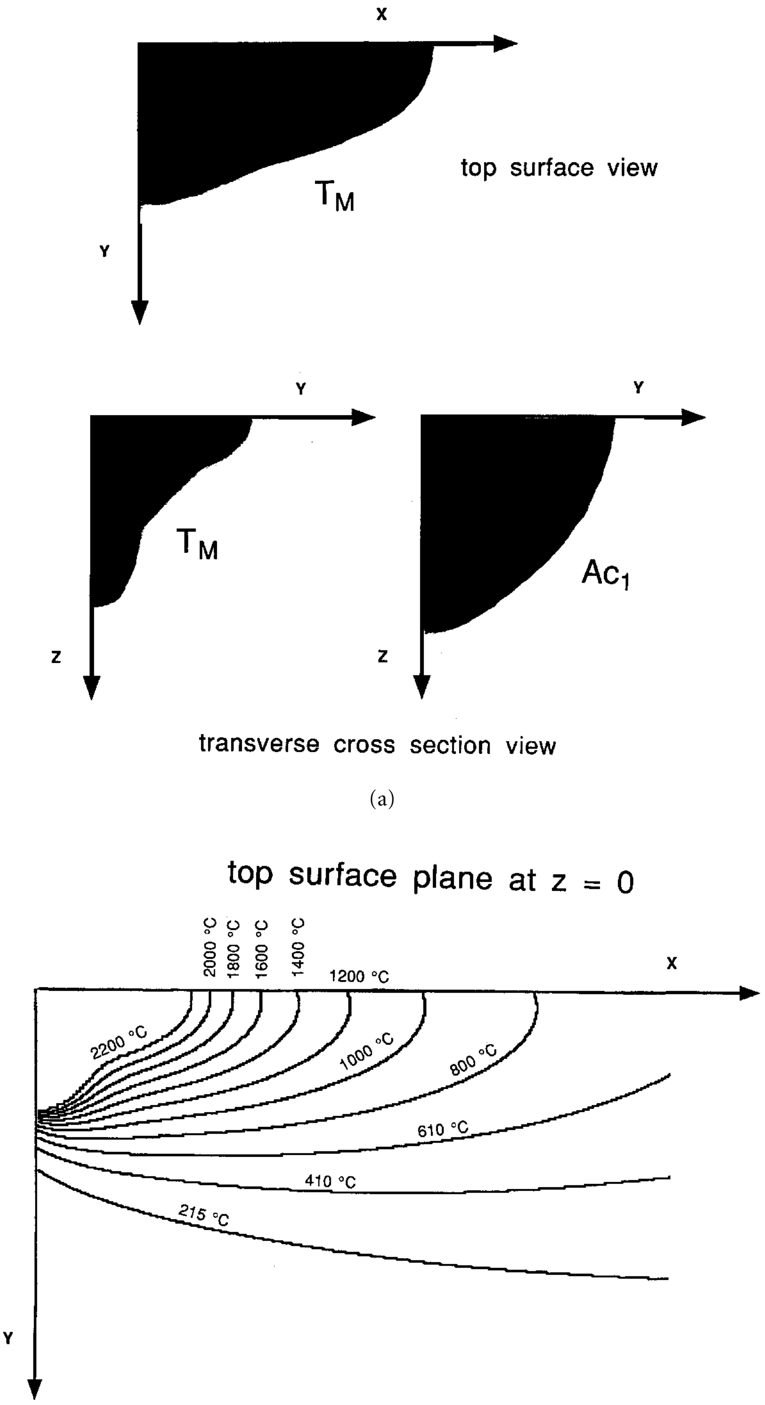

point sources of energy
subdomains
upstream boundaries of subdomains
downstream region of solution domain consisting of one or more subdomains

FIGURE 6.12 Schematic representation of a pseudo steady state temperature field illustrating the equivalence between a given finite distribution of subdomains defined by constrained surface values of temperature and an energy source defined by a given power and spatial distribution.

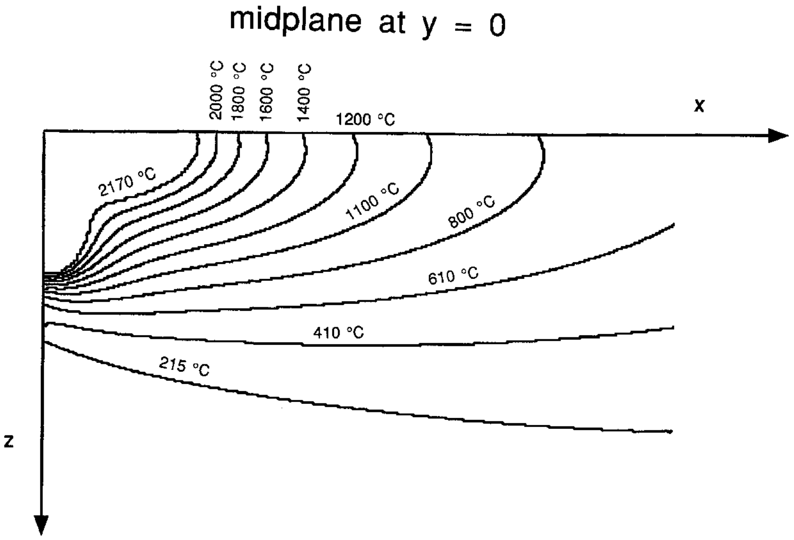
is an equal weight of the summed quantities by all the boundaries defining the subdomain. Discretization errors resulting from stiffness over regions near the upstream boundaries are reduced because of a propagation of information concerning the gradient from downstream boundaries. Second, a partitioning of the solution domain via a given set of distributed subdomains is equivalent to specifying an energy source of a given power and spatial distribution. This equivalence is illustrated in Fig. 6.12. Referring to this figure, we consider a partitioning of the region defined by the pseudo steady state of the weld into two regions, one which is upstream and defined mostly by the spatial distribution of the energy source and another which is downstream and defined mostly by the total volume of the weld but excluding that part occupied by the energy source. Shown in the downstream section of Fig. 6.12 is a representation of a closed and finite sized subregion which is defined by an upstream boundary and a set of isothermal surfaces located downstream. The temperature field within this subregion is assumed to have been generated by two point sources of energy located on the surface of the workpiece. The isothermal surfaces partition the solution domain into three subdomains. The equivalence between the energy sources and domain partitioning is illustrated by noting that for a given domain partitioning defined by an upstream boundary upon which there is a given temperature distribution and two isothermal surfaces of given temperatures and relative spatial extent, one can find two point energy sources whose relative separation and magnitude are such that they generate the given set of subdomains. By superposition, it follows that this equivalence can be extended to the more general case of volumetric deposition of energy and of temperature dependent material properties. A general restriction on domain partitioning, however, follows from this equivalence. It is possible to adopt a specific partitioning of the solution domain that does not correspond to a physically realistic energy source. Considering this, it then follows that in the absence of any constraints based on experimental information concerning the three-dimensional character of the temperature field (e.g., the case considered here) the generating function $T_E(x)$ must be sufficiently accurate for generating a partitioning of the solution domain which is physically realistic. It should be noted that this does not represent an inconvenient restriction in practice on the construction of $T_E(x)$.



(a)

(b)

FIGURE 6.13 Cross sections of estimated temperature field calculated by means of generating function with the condition that the isothermal surfaces at T_M and *Ac*1 are consistent with the experimental cross section shown in Fig. 6.7. Shown in Fig. (6.13a) are transverse cross sections for the T_M and *Ac*1 boundaries and a top surface profile of the T_M boundary all of which have been calculated by means of a generating function. The temperature fields shown in Figs. (6.13b) and (6.13c) are those on the top surface and symmetry planes, respectively, of the model system.



(c)

FIGURE 6.13 (continued.)

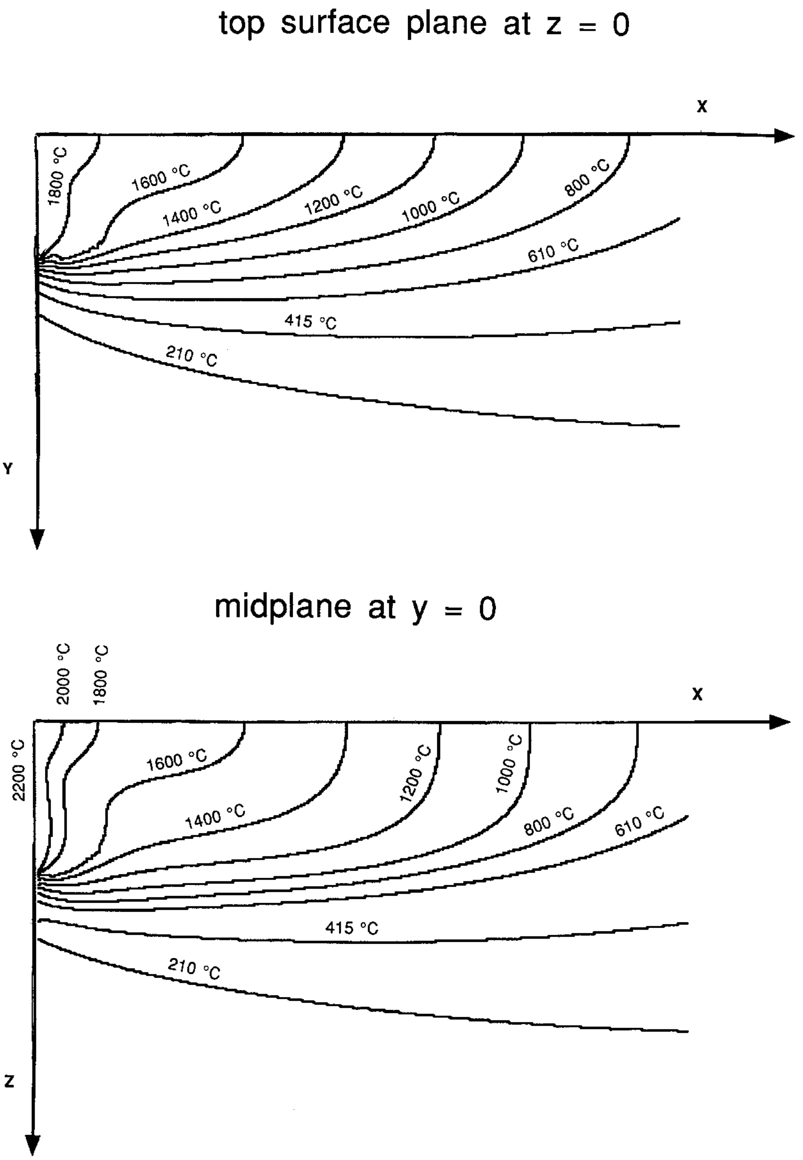
Prototype case study 2. Optimization of discrete temperature field based on multiple constraints—For the present case study we consider a prototype analysis of the weld whose cross section is shown in Fig. 6.8 and whose process parameters are given in data set 2 of Appendix A. For this analysis we adopt three separate measurements as constraints. These are the transverse cross section of the solidification boundary, the solidification boundary at the top surface, and the transverse cross section of the Ac_1 isotherm. For this analysis we adopt the same optimization procedure applied in the case study presented above.

The first stage of our analysis for this case entails the generation of a three-dimensional isothermal surface at T_M whose transverse projection maps onto the experimentally measured transverse cross section of the solidification boundary and whose intersection with the top surface of the workpiece assumes the shape of the experimentally measured solidification boundary at the top surface. In addition, we generate a three-dimensional isothermal surface whose transverse projection maps onto the experimentally measured Ac_1 cross section.

Shown in Fig. 6.13a are two-dimensional cross sections of a temperature field, calculated via a generating function $T_E(x)$, for which the isothermal surfaces at temperatures T_M and Ac_1 are consistent with Fig, 6.8. The assumed values of T_M and Ac_1 are given in appendix A. A feature of this calculation that should be noted is that we have adopted a generating function which does not represent any type of reasonable estimate of the temperature field. For this calculation we have arbitrarily adopted a generating function which is not symmetric with respect to the midplane boundary.

The next stage of this analysis consists of partitioning the solution domain into subdomains. A temperature field obtained via a function $T_E(x)$ which satisfies this partitioning is shown in Fig. 6.13b. It is significant to note that for this calculation two of the constrained boundaries are used for embedding known information about the weld, in contrast to the previous analysis, where only one boundary served this purpose.

In the next stage of this analysis the objective function Eq. (6.32) is minimized with respect to adjustment of the distribution of temperatures on the upstream boundary. Shown in Fig. 6.14 are values of the temperature field which have been calculated by minimizing the objection function Eq. (6.32) for a specific distribution of temperatures on the upstream boundary and the subdomain partitioning defined by the temperature field shown in Figure 6.13. Again, as in the analysis presented in the previous section,



top surface plane at z = 0

midplane at y = 0

FIGURE 6.14 Cross-sections of discrete temperature field whose values have been optimized with respect to a given set of upstream boundary values and constraint conditions on the isothermal surfaces T_M and $Ac1$ defined by the generating function described in Fig. 6.13.

the calculation shown in this figure represents a partial optimization in the sense that the objective function assumes values reasonably close to (but strictly not at) its minimum value. This follows since the solution domain does not extend over the entire temperature field of the pseudo steady state of the weld.

Some significant features of this calculation are now examined. For the calculated temperature field shown in Fig. 6.14, the solution-domain partitioning is based on *a priori* information concerning the weld which includes quantitative aspects of its three dimensional character, in contrast to the type of *a priori* information used in the preceding analysis which is relatively more qualitative. In particular, for the preceding analysis, partitioning of the solution domain is undertaken by means of a generating function $T_E(x)$ whose general character is qualitatively consistent with that of a temperature field that would be generated from a physically realistic energy source. The generation of a physically realistic distribution of subdomains, as well as the generation of a good estimate of the temperature field throughout the solution domain, does not represent a difficult-to-achieve condition on the construction of $T_E(x)$. Only some general quantitative knowledge concerning the general three dimensional trend of the temperature field is required for any realistic specification of $T_E(x)$. In addition, for cases where there is not a large amount of *a priori* information concerning the three-dimensional character of the temperature field, the discrete temperature field can be optimized with respect to the relative distribution of subdomains as well as the specified constraint conditions on the system. In the present analysis, however, partitioning of the solution domain is based mostly on quantitative experimental measurements concerning the three-dimension character of the temperature field. In particular, because two of the constrained boundaries are used for embedding information, this analysis represents a constrained optimization of the discrete temperature field using constraints related to experimental information concerning the values of the temperature as well as of the temperature gradient.

Prototype case study 3. Embedded mesh procedure based on constrained partitioning of solution domain—For the present case study we consider the analysis of a weld for which thermal histories are desired that extend over a relatively large range of temperatures, i.e., from relatively high temperatures at regions close to the energy source to relatively low temperatures that are close to the ambient temperature of the workpiece. For the present case most of the temperature field associated with the pseudo-steady state consisting of values above the ambient temperature are within the defined solution domain. This implies that the last term in Eq. (6.14) is relatively small and that (to the extent that one has confidence in the value of q_o/V_B, optimization of the temperature field can be achieved at a minimum value of the objective function that is close to zero.

The first stage of this analysis, as in general, is the partitioning of the solution domain into a set of distributed subdomains upon whose boundaries are constrained temperature values T_E which are consistent with *a priori* knowledge of the weld. For this case study, we have assumed that there is sufficient *a priori* information concerning the weld that the specified solution domain, and associated distribution of subdomains, extends over most of the temperature field that exceeds the ambient temperature of the workpiece but is not within the region containing the energy source. It is assumed that based on cross section and thermocouple measurements, a distribution of subdomains is constructed, by means of a generating function $T_E(x)$, such that the isothermal surfaces at T_M, 850°C and 430°C are constrained. In practice, however, a relatively more dense distribution of subdomains can be adopted which tracks more closely the distribution of thermocouple measurements.

Shown in Figs. 6.15 through 6.19 are a series of two-dimensional cross sections of a temperature field that has been calculated by means of a two stage procedure. The first stage consists of adopting a generating function $T_E(x)$ that provides both a means of partitioning the solution domain and an initial estimate of the temperature field throughout the solution domain. The second stage consists of the constrained optimization of the discrete temperature field over the solution domain according to a given set of values T_B on the upstream boundary and a specified set of top surface and midplane boundary conditions.

Some significant features of this calculation are now examined. The constrained optimization of the discrete temperature field over the entire solution domain consists of separate optimizations over the individual subdomains. These optimizations are not independent, however, because they are linked through the distribution of boundary values T_B on the upstream boundary of the solution domain. Accordingly, it is seen that the discrete temperature field within a given subdomain is determined by the local distribution of values T_B on the upstream boundary of the solution domain which is part of the closed surface of the given subdomain. It follows that by optimizing T_B with respect to q_O/V_B one is able to deduce information concerning the spatial distribution of temperature on the upstream boundary, or equivalently, information concerning the spatial characteristics of the energy source. Inversely, any *a priori* information concerning the spatial characteristics of the energy source can be adopted as constraints on the temperature field T_B. As discussed in the previous section, a distributed set of subdomains provides a means of embedding information concerning the gradient of the temperature field. For a given subdomain, however, there is an upper bound on the value of the temperature gradient. This property provides the basis for varying the

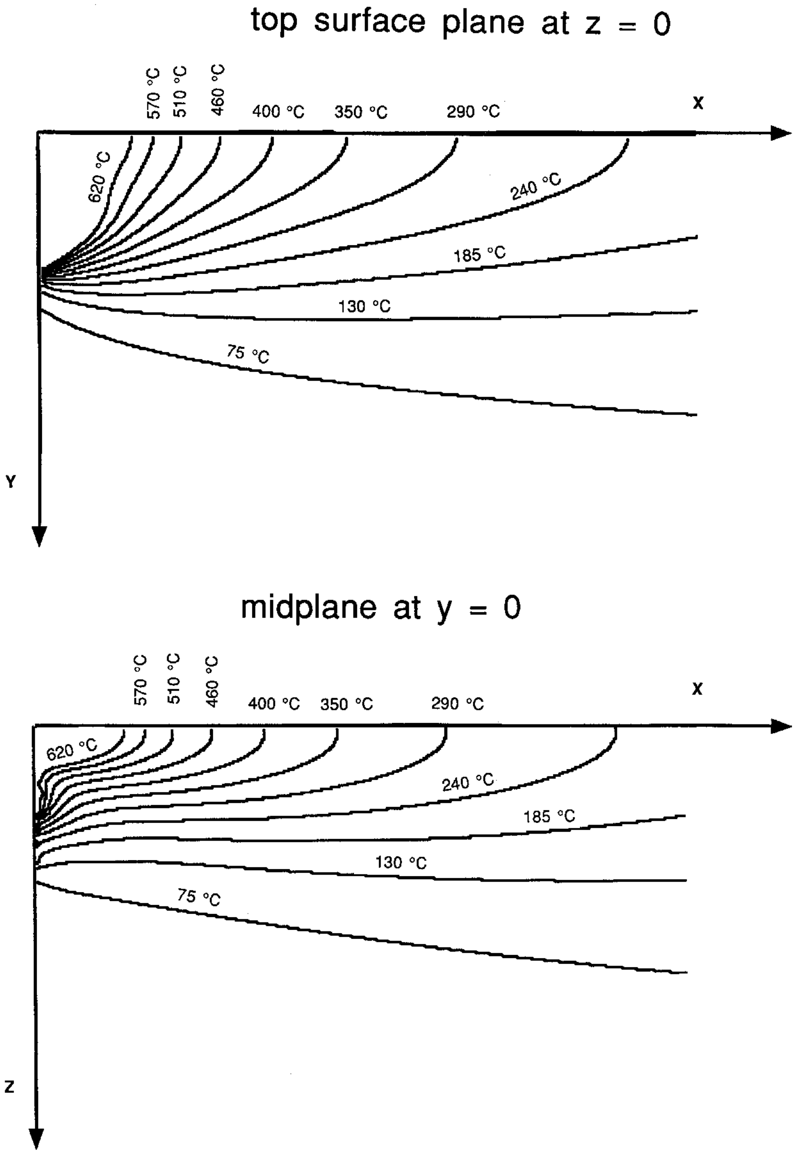


FIGURE 6.15 Cross-sections of estimated temperature field calculated by means of a generating function extending over the entire solution domain. Discretization of temperature field is not uniformly distributed throughout solution domain.

level of discretization of the temperature field depending upon characteristic maximum of the temperature gradient associated with a given subdomain. That is to say, this property provides the basis for a discretization of the temperature field by means of a nested set of embedded meshes of progressively finer resolution. As an example, consider the constrained optimization leading to the discrete temperature field whose two dimensional cross sections are shown in Figs. 6.15 through 6.19 is with respect to a set of embedded meshes whose discretization lengths are $4\Delta l$, $2\Delta l$, and Δl, where $\Delta l = 0.2$ mm.

Shown in Fig. 6.15 are two-dimensional cross-sections of an estimated temperature field calculated by means of a generating function $T_E(x)$. For this calculation the generating function consists of a sum of Rosenthal solutions of the type given by Eq. (6.B1) in Appendix B. The energy per distance corre-

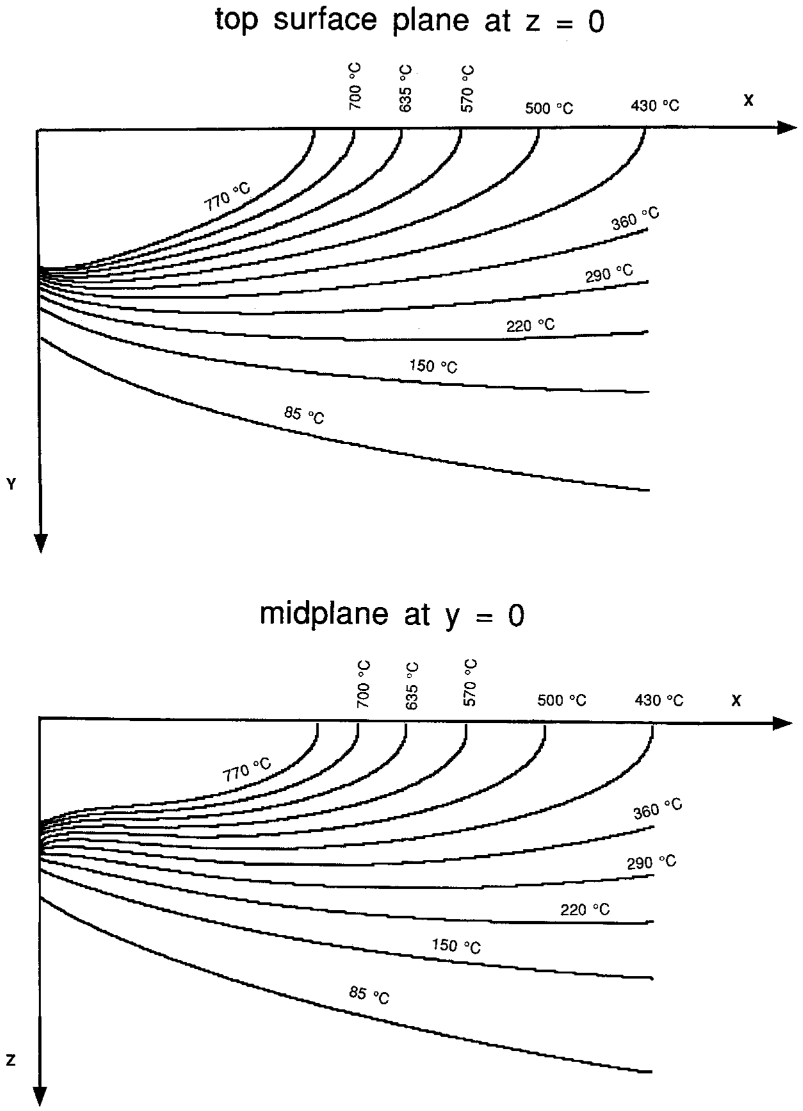


FIGURE 6.16 Cross-sections of estimated temperature field over calculated by generating function extending mostly over subdomains whose temperatures are less than 430°C.

sponding to this estimated of the temperature field is 4.56 × 10^6 J/m. Figs. 6.16 and 6.17 describe the process of optimizing the discrete temperature field over a subdomain whose closed surface is defined by the upstream boundary of the solution domain and isothermal surfaces at 430°C and 850°C. The discretization of this subdomain is with respect to a mesh of 0.4 mm. Figures 6.18 and 6.19 describe a similar optimization process where the discretization of the subdomain is with respect to a mesh of 0.2 mm. The closed surface of this subdomain is defined by a section of the upstream boundary of the solution domain and an isothermal surface at 850°C. In addition, for this subdomain, the isothermal surface at T_M is constrained. The combined temperature field consisting of the estimated temperature field whose values are below 430°C, obtained by means of a generating function $T_E(x)$, and the temperatures fields obtained by constrained optimization over the two nested subdomains, represented by Figs. 6.17 and 6.19, corresponds to energy per distance of 4.27 × 10^6 J/m.

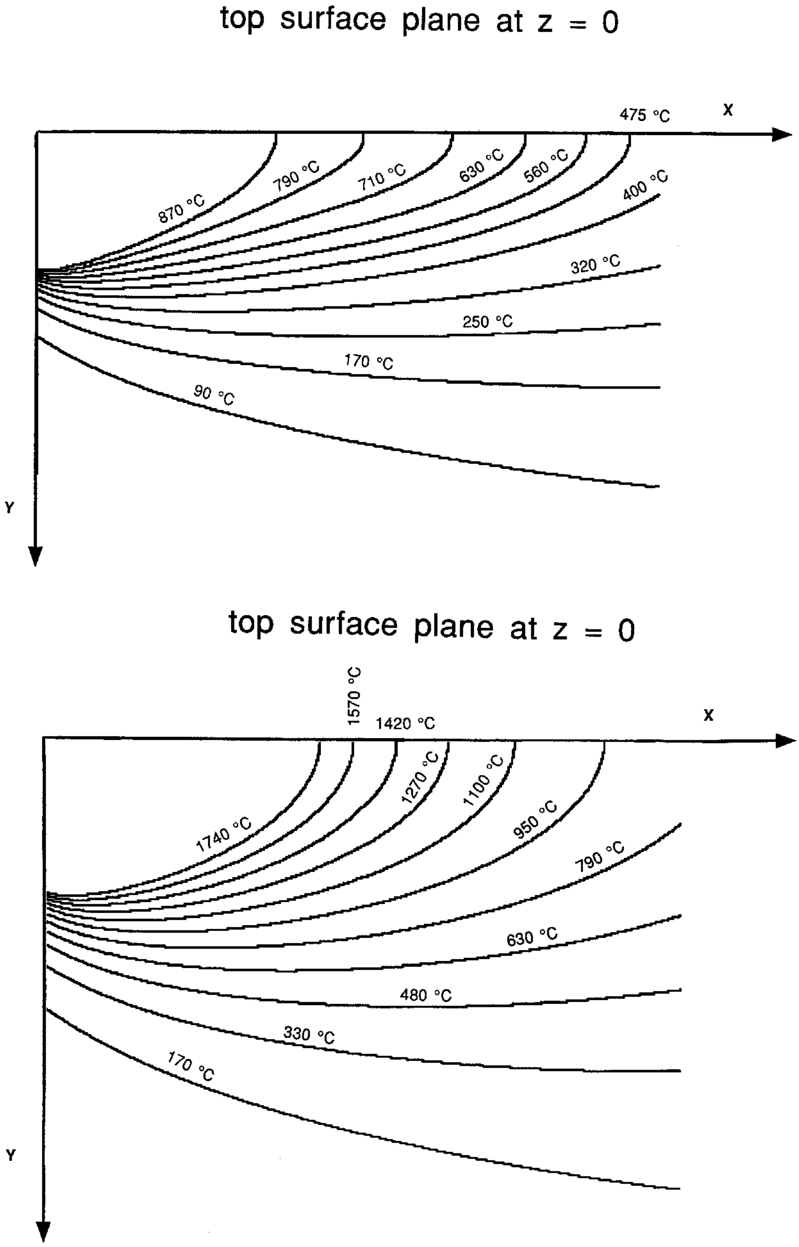
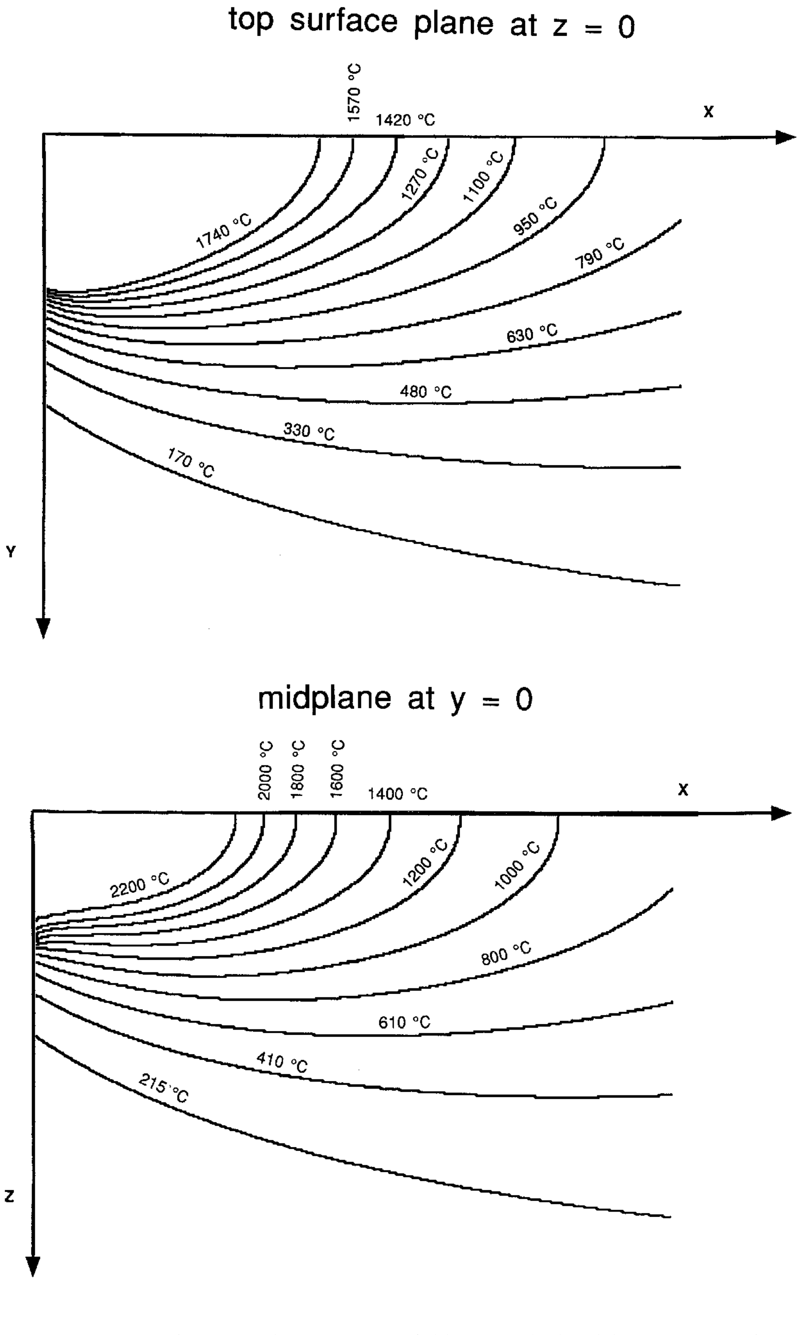


FIGURE 6.17 Constrained optimization of discrete temperature field, with respect to a mesh resolution of 0.4 mm, using estimated field shown in Fig. 6.16 and constraint conditions on isothermal surfaces at 430°C and 850°C.

6.7 Implicit Representation of Temperature Dependence of Material Properties

In this section we discuss a specific property of constraints which is significant in that it provides a means for including implicitly information concerning the temperature dependence of material properties as well as providing the basis for a reformulation of the problem in terms of the solution of an elliptic equation defined over a closed domain. With respect to this property, the interesting aspect of a reformulation



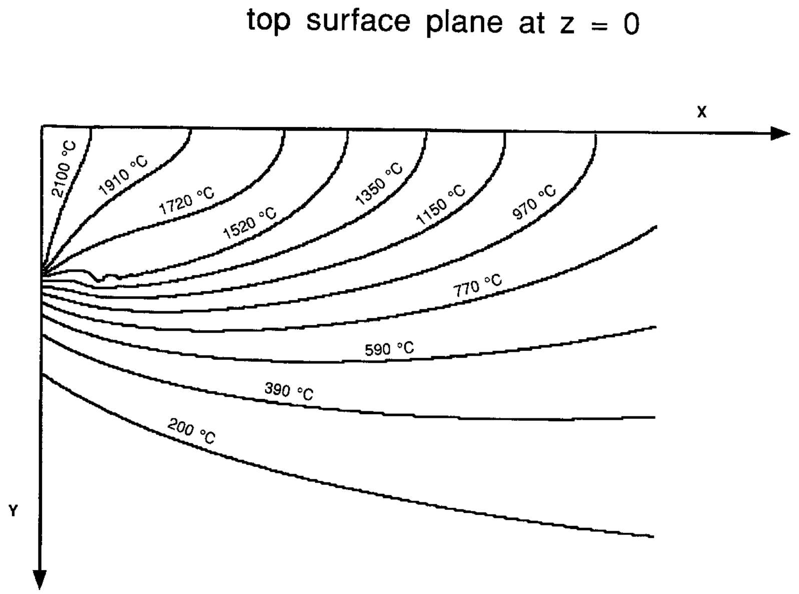
top surface plane at z = 0

midplane at y = 0

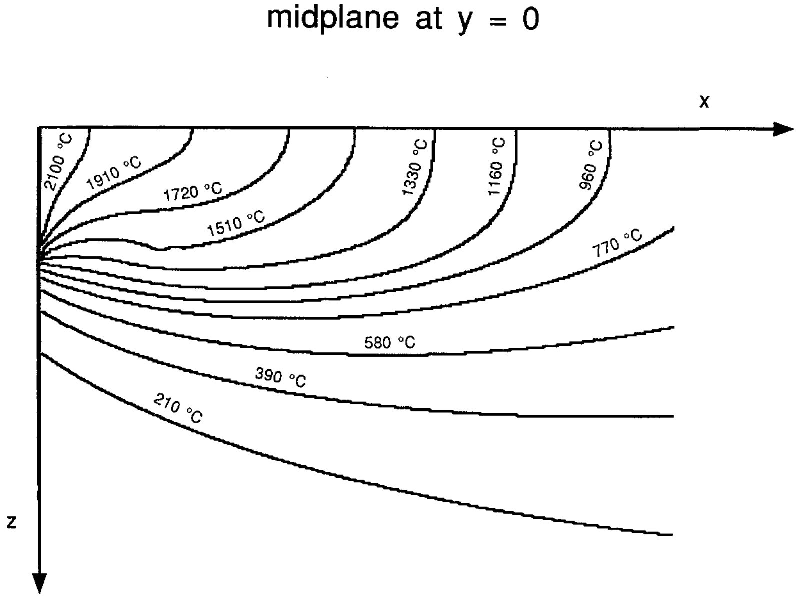
FIGURE 6.18 Cross-sections of estimated temperature field over calculated by generating function extending mostly over subdomains whose temperatures are less than 850°C.

in terms of an elliptic-equation representation is that, rather than adopting a representation such as Eq. (6.31) which is based on a generalized diffusivity, it is possible to adopt a representation based on the solution of the Laplace equation

$$\nabla^2 T = 0 \tag{6.33}$$



top surface plane at z = 0



midplane at y = 0

FIGURE 6.19 Constrained optimization of discrete temperature field, with respect to a mesh resolution of 0.2 mm, using estimated field shown in Fig. 6.18 and constraint conditions on isothermal surfaces at T_M and 850°C.

defined over a reasonably distributed set of closed subdomains upon whose boundaries constraint information is specified. As is obvious, a reformulation of the problem in terms of Eq. (6.33) should provide relatively more convenience computationally.

An examination of the mathematical foundation of posing the problem as a solution of Eq. (6.33) defined over a distributed set of subdomains, while consistently representing implicitly the temperature dependence of material properties, follows from Eq. (6.27). This examination is in terms of an illustration of the interrelationship between two alternative discrete representations of the problem. One representation is derived from a parabolic equation while the other is from an elliptic equation. We first consider

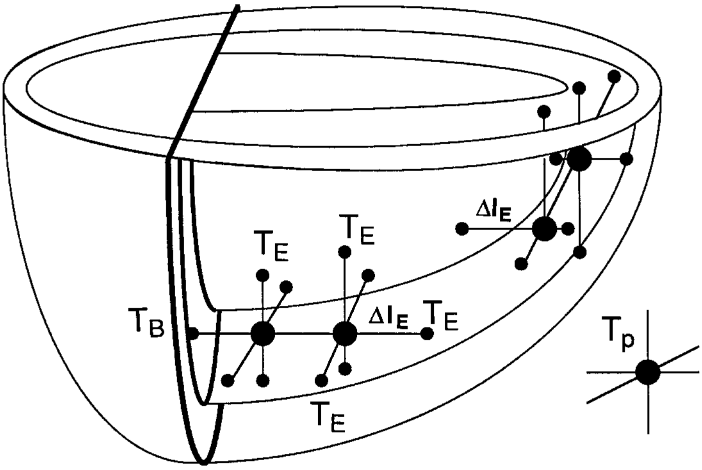


FIGURE 6.20 Schematic representation of a finite distribution of subdomains providing implicit information concerning the temperature dependence of material properties and of the coupling between energy transfer and fluid convection.

the process of calculating a discrete temperature field T_p over an open solution domain via numerical integration of Eq. (6.27) for a given set of upstream boundary conditions . Given an explicit function $\kappa(T)$, a discrete set of upstream boundary values T_B and specified top surface and midplane boundary conditions, a unique discrete temperature field T_p is determined over an open solution domain according to Eq. (6.27). Next, we consider that within this same open solution domain we arbitrarily define a set of closed subdomains such as those shown schematically in Fig. 6.12. Referring to this figure, we next consider the process of calculating a discrete temperature field T_p over a given closed subdomain for a given set of boundary conditions via an iterative procedure. Given an explicit function $\kappa(T)$ and boundary conditions over the entire surface of the closed domain, a unique discrete temperature field T_p is determined according to Eq. (6.27). Thus it is seen that depending on the type of boundary conditions, Eq. (6.27) can assume either the role of an elliptic or that of a parabolic solver. Considering again a given set of closed surfaces such as are shown schematically in Figure 6.12, and following the set of different types of temperature-field designations defined in previous sections, we represent values of the temperature field on the downstream portion of this surface by the quantity T_E. At this point, it is significant to observe that if the temperature field T_E is interpreted as being part of the temperature field T_p determined via Eq. (6.27), adopted as a parabolic solver, then it follows that the quantity T_E can be interpreted as being defined by a function

$$T_E = T_E(T_B, \kappa, V_B). \tag{6.34}$$

Next, we adopt the condition that there is sufficient information concerning the temperature field downstream (e.g., information that has been obtained from weld cross sections or thermocouple measurements) that a specified closed solution domain can be partitioned into a set of distributed subdomains such as are shown in Fig. 6.20.

For a given partitioning of the solution domain into a set of distributed subdomains (e.g., Fig. 6.20), we consider the process of determining a discrete temperature field T_p over a closed solution domain by adopting Eq. (6.27) as an elliptic solver and the values of temperature on the subdomain surfaces as constraints. We wish to show that for a distributed set of closed subdomains, the equality

$$T_p = \frac{1}{6w_p^*}\sum_{i=1}^{6} w_i^* T_i = \frac{1}{6}\sum_{i=1}^{6} T_i \tag{6.35}$$

is satisfied because constrained values of the field quantity T_E represent an implicit inclusion of information concerning the temperature dependence of the material properties and of the motion of the workpiece relative to the energy source. The representation defined by Eq. (6.35) is the discrete analog of the Laplace equation and implies that within each subdomain the temperature field is determined via Eq. (6.33).

Referring to Fig. 6.20, it follows that by linear interpolation in three-dimensions, the discrete temperature field T_p is given by

$$T_p = \sum_{i=1}^{6} W_{E,i} T_{E,i}, \tag{6.36}$$

where

$$W_{E,i} = \frac{1}{\Delta l_{E,i}} \left(\sum_{m=1}^{6} \frac{1}{\Delta l_{E,m}} \right)^{-1} \tag{6.37}$$

and the discrete quantities $\Delta l_{E,i}$, shown in Fig. 6.20, satisfy the conditions

$$\sum_{i=1}^{6} W_{E,i} = 1 \quad \text{and} \quad \Delta l_{E,i} \geq \Delta l \tag{6.38}$$

where the has been partitioned into a set of discrete volume elements of volume $(\Delta l)^3$. The quantities $\Delta l_{E,i}$ ($i = 1,\ldots, 6$) are the distances between a given discrete location within a subdomain (indexed by the integer variable p) and its six discrete nearest neighbor locations (indexed by the integer variable i) which are on the closed surface of the subdomain.

Equation (6.36) represents a weighted average of the temperature field T_E over the closed surface of a subdomain. It is significant to note that the general form of the sum defined by Eq. (6.36) implies that Eq. (6.35) is an appropriate representation of the discrete temperature field within a closed subdomain. A semi-quantitative proof showing that the existence of the sum defined by Eq. (6.36) implies that the equality represented by Eq. (6.35) is satisfied within a closed subdomain is given in Appendix D. Our proof, which is based on Green's theorem,[8] is in terms of discrete analogs of mathematical representations based on continuous functions and operators. A qualitative and intuitive understanding of the equality expressed by Eq. (6.35) follows from the observation that the right side of this equality, applied iteratively over a closed domain, is simply an interpolation among the surface values T_E that bound that domain.

6.8 Implicit Representation of Coupling Between Heat Transfer and Fluid Convection

In this section we discuss a specific property of constraints which provides a means for including information concerning the coupling between heat transfer and fluid convection. This property is related to that examined in the previous section and provides a basis for posing the weld thermal-history problem in terms of the solution of an elliptic equation (i.e., the Laplace equation Eq. (6.33)) defined over a closed domain. The mathematical foundation for posing the problem as a solution of Eq. (6.33) over a distributed set of subdomains, and consistently representing the coupling between heat transfer and fluid convection, follows from the formal structure of the discretization procedure employed by the SIMPLE algorithm.[9] With respect to the structure of this discretization procedure, we first consider the process of calculating a discrete temperature field T_p over an infinite solution domain for a given set of upstream boundary conditions and a specified flow field U representing the flow of liquid in the weld meltpool.

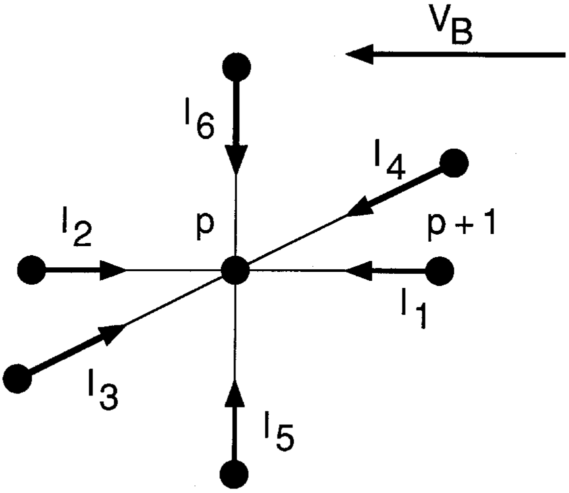


FIGURE 6.21 Indexing scheme for set of quantities defined in Eq. (6.40).

For the purpose of our analysis we adopt a specific representation of the discretization employed by the SIMPLE algorithm which is defined according to Fig. 6.21. Accordingly, it follows that given an explicit function $\kappa(T)$, a discrete set of upstream boundary values T_B, specified top surface and midplane boundary conditions, and a three-dimensional flow field U defined over the region of the meltpool, a unique discrete temperature field is determined according to the weighted sum

$$T_p = \frac{1}{6a_p^*}\sum_{i=1}^{6} a_i T_i, \tag{6.39}$$

where

$$a_i = \kappa_i \max\left[\left(1.0 - 0.1\left|\frac{(\Delta l)(U \cdot I_i)}{\kappa_i}\right|\right), 0\right] + (\Delta l)\max[U \cdot I_i, 0] + \delta_{i,p+1} V_B \Delta l \tag{6.40}$$

and

$$a_p^* = \frac{1}{6}\sum_{i=1}^{6} a_i. \tag{6.41}$$

Next, by adopting arguments similar to those in the previous section, it follows that the temperature field T_E associated with an arbitrary downstream surface (e.g., Fig. 6.20) can be interpreted as being defined by a function

$$T_E = T_E(T_B, \kappa, V_B, U) = T_E(T_B, \kappa, \rho, \mu, \partial\gamma/\partial T, \beta). \tag{6.42}$$

The quantities ρ, μ, $\partial\gamma/\partial T$, and β are the density, coefficient of viscosity, surface tension coefficient, and coefficient of expansion, respectively, of the liquid metal. The dependence of the flow field U on these quantities is through the Navier-Stokes and Continuity equations. As in the previous section, we consider a partitioning of the solution domain into a set of distributed subdomains and the process of determining a discrete temperature field T_p over a closed solution domain by adopting Eq. (6.39) as an elliptic solver and the values of temperature on the subdomain surfaces as constraints. It follows from Eq. (6.39) that for a distributed set of subdomains the equality

$$T_p = \frac{1}{6a_p^*}\sum_{i=1}^{6} a_i T_i = \frac{1}{6}\sum_{i=1}^{6} T_i \tag{6.43}$$

is satisfied. Adopting the same reasoning presented in the previous section, the equality expressed by Eq. (6.43) follows because the constrained values of the field quantity T_E represent an implicit inclusion of information concerning the coupling between energy transfer and fluid flow. Again we note that the equality expressed by Eq. (6.43) represents the discrete analog of the Laplace equation and implies that over each subdomain the temperature field is determined via Eq. (6.33). A semi-quantitative proof of Eq. (6.43) (the same as that for Eq. (6.35)) is given in Appendix D.

6.9 Conclusion

A general overview of the method of geometric constraints has been presented. A series of case studies have been presented for the purpose of demonstrating the flexibility of this method. Our presentation of the general features of the geometric constraints method has been strictly by means of illustrating concepts rather than in terms of rigorous mathematical developments. The mathematical foundation of the geometric constraints method is that of constrained optimization of a discrete parameter system. Following this method, the solution is driven equally from both upstream and downstream boundaries, in contrast to the standard approach of associating all parameterization with upstream regions of the weld. The problem is posed as a discrete optimization problem over a closed and finite domain where all boundaries are given equal weighted and where boundary values, both upstream and downstream, represent the input of information into the model system. The formal mathematical distinction of our approach is that it casts the dynamic weld problem terms of the solution of an elliptic equation. Although our approach is based on parameter optimization in the strict sense, it does not concern the optimization of any parameters associated with a physical model representation of the dynamic weld. In general, no parameterization of the model system is assumed beyond that of a finite set of temperatures, conductivities and diffusivities, i.e., $\{T_p, k_p, \kappa_p\}$ where $p = 1, \ldots, N_S$. Any parameterization associated with the generating functions, $T_B(x)$ or $T_E(x)$, represents only an estimate and must not be interpreted as either a parameterization of the system or a characterization of the energy source. Accordingly, the power input q_0 is not to be interpreted as a parameter but rather as a constraint.

We have not considered in the course of our discussions or case studies two major aspects of complex welding processes that are adaptable to an analysis of welds using the method of geometric constraints. These are detailed information concerning the energy source and information concerning relatively complex shapes of the pseudo steady state of the weld.[15,16] Detailed information concerning the energy source would be related to constrained optimization of the third term in Eq. (6.14) through the specification of the discrete quantities ΔT_S defined in the set relation defined by Eq. (6.8). One of the origins of complex weld shapes is the specific type of process employed for energy and filler metal deposition which can cause the meltpool to assume a configuration which extends above the top surface of the workpiece. Another origin of complex weld shapes is the workpiece geometry which can consist of irregularly shaped grooves such as those occurring in multipass welding processes.

Finally, we have not considered in the course of our discussion the fact that, in principle, explicit model representations of energy sources can be combined with the geometric constraints method. A procedure of this type would be such that parameters associated with a given explicit model representation of an energy source are optimized with respect to a given set of constraint conditions, in addition to a discrete temperature field T_p. This procedure would represent a direct generalization of the procedure described in this report which involves adjusting the distribution of values of the quantity T_B on the upstream boundary of the solution domain.

Acknowledgments

The authors would like to thank Dr. J.O. Milewski of the Materials Science and Technology Division, MST-6, Los Alamos National Laboratory, and Dr. Milo Kral of the Naval Research Laboratory for their comments during the preparation of this manuscript.

References

1. S.G. Lambrakos, E.A. Metzbower, J. Milewski, G. Lewis, R. Dixon and D. Korzekwa, *J. Materials Eng. Performance,* 3(5), 639 (1994).
2. S.G. Lambrakos and J. Milewski, Analysis and Refinement of a Method for Numerically Modelling Deep Penetration Welding Processes using Geometric Constraints, *J. Materials Eng. Performance,* 4(6), 717 (1995).
3. O. Grong, *Metallurgical Modelling of Welding,* Institute of Materials, London (1994).
4. G.R. Walsh, *Methods of Optimization,* John Wiley & Sons, London (1975).
5. C.S. Beightler, D.T. Phillips, and D.J. Wilde, *Foundations of Optimization,* 2nd ed., Prentice-Hall International series in Industrial and Systems Engineering, Prentice-Hall, New Jersey (1979).
6. M.R., *Finite Algorithms in Optimization and Data Analysis,* Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons, New York(1985).
7. H.S. Carslaw and J.C. Jaeger, *Conduction of Heat in Solids*, 2nd ed., Oxford (1959).
8. J.D. Jackson, *Classical Electrodynamics,* John Wiley & Sons, New York (1962) (see p 15 for discussion concerning elliptic and parabolic equations).
9. S.V. Patankar, *Numerical Heat Transfer and Fluid Flow,* Hemisphere Publishing (1980).
10. D. Rosenthal, *Weld. J. Res. Suppl.,* 20, 220 (1941).
11. D. Rosenthal, *Trans. ASME,* 68, 849 (1946).
12. R.C. Reed and H.K.D.H. Bhadeshia, Acta Metall. Mater., (42)11, 3663 (1994).
13. T. Zacharia and S.A. David, *Mathematical Modelling of Weld Phenomena,* H. Cerjack and K.E. Easterling, Eds., Institute of Materials, London (1993). A relatively complete collection of material properties is given for 304 stainless steel.
14. D. Zwillinger, *Handbook of Differential Equations,* Academic Press, New York (1989) (see p 514, 579).
15. *The Procedure Handbook of Arc Welding,* 13th ed., The Lincoln Electric Company, Cleveland, OH, (1994).
16. J.F. Lancaster, *The Physics of Welding,* 2nd ed., International Institute of Welding, Pergamon Press, New York (1986).

Appendix A

Parameters and Physical Quantities

$Ac_1 = 675.0$ C $\quad Ac_3 = 840.0$ C $\quad T_M = 5020.0$
$k = 25$ W/mC $\quad \rho C_p = 5 \times 10^6$ J/m^{3}C $\quad \kappa = 5 \times 10^{-6}$ m^2/s

Welding Parameters for Data Set 1:

Type: bead on plate with reinforced top surface
Weld speed V_B: 3.61×10^{-3} m/s
Average energy input Q/D: 2.298×10^3 KJ/m
Bead depth below surface: 6.25 mm
Bead width at surface: 19 mm
Plate thickness: 0.0508 m
Power input $(Q/D)V_B$: 8.271×10^3 W
Grid spacing: $\Delta l = 1 \times 10^{-4}$ m
Time interval for passage of discrete volume
elements past a given point: $\Delta t = \Delta l/V_B = 5.540 \times 10^{-2}$ s

Welding Parameters For Data Set 2:

Type: bead in groove with flat top surface
Weld speed V_B: 3.81×10^{-3} m/s
Average energy input Q/D: 2.159×10^{3} kJ/m
Bead depth below surface: 9.5 mm
Bead width at surface: 15 mm
Plate thickness: 0.0254 m
Maximum length of trailing solidification boundary at surface: 14.5 mm
Power input $(Q/D)V_B$: 8.226×10^{3}
Grid spacing: $\Delta l = 2 \times 10^{-4}$ m
Time interval for passage of discrete volume elements past a given point: $\Delta t = \Delta l / V_B = 2.625 \times 10^{-2}$ s

Appendix B

Derivation of Weighted Finite-Sum Representation of Pseudo Steady State Based on Rosenthal Solution

According to the Rosenthal solution of Eq. (6.1) for thick plate boundary conditions, the change in temperature at a position p due to an energy source at position $p + 1$ whose power is q_{p+1}, distance from p is R and speed relative to the workpiece is V_B is

$$\Delta T_p = \frac{q_{p+1}}{2\pi k_p R} \exp\left[-\frac{V_B}{2\kappa_p'}(R + x)\right], \tag{6.B1}$$

where

$$\kappa_p' = \frac{k_p}{\rho(T_p)C_p(T_p)}. \tag{6.B2}$$

For a uniform discretization of the solution domain corresponding to a node separation of Δl, it follows that

$$q_{p+1} = \frac{\rho(T_p)C_p(T_p)(T_p - T_{p+1})(\Delta l)^3}{\Delta t}, \tag{6.B3}$$

$R = \Delta l$, $x = \Delta l$ and $\Delta y = \Delta l / V_B$. Next, we note that for a discrete partitioning of the temperature field, a change in temperature at position p is related to the changes in temperature of its nearest neighbor positions by the expression

$$\Delta T_p = \frac{1}{6\kappa_p} \sum_{i=1}^{6} \kappa_i \Delta T_i, \tag{6.B4}$$

where $\Delta T_i = T_i - T_p$. Equating Eq. (6.B1) and Eq. (6.B2), it follows that

$$\frac{1}{6\kappa_p} \sum_{i=1}^{6} \kappa_i (T_i - T_p) = \frac{(T_p - T_{p+1}) V_B \Delta l}{2\pi\kappa_p'} \exp\left[-\frac{V_B \Delta l}{\kappa_{p'}}\right] \tag{6.B5}$$

Next, adopting the approximations $\kappa_p' = \kappa_p$ and

$$\frac{6}{2\pi}\exp\left[-\frac{V_B\Delta l}{\kappa_p'}\right] \approx 1, \tag{6.B6}$$

Eq. (6.27) follows from Eq. (6.B5). Note that although the present derivation is based on the Rosenthal thick plate solution any Rosenthal-type solution is sufficient since we are considering only highly localized energy transfer. For this case and any given Rosenthal-type solution, any additional terms that would be associated with either the relative position of the energy source within the workpiece or the workpiece geometry will be negligible for sufficiently small Δl.

Appendix C

Some Properties of Convex Functions that are Relevant to the Construction of Objective Functions

Our choice of the particular forms of the objective function presented in this report is arbitrary and primarily representative except for the criterion that these functions be convex. Therefore with respect to the general analysis presented here questions concerning the best form of the objective function remain open. Two properties of convex functions, however, which are relevant to the construction of objective functions whose minimum values correspond to an optimal discrete temperature field of the pseudo steady state of the weld, are as follows.

1. A convex function has the property that its value at an interpolated point is less than or equal to the value that would be obtained by linear interpolation.
2. The sum of a finite number of convex functions is itself a convex function.

The first property is significant with respect to our development because the finite weighted sums underlying the general formulation of the geometric constraints method represent linear interpolations over a given interval Δl. Convexity of an objective function implies, therefore, that errors associated with discretization of the temperature field (where this field tends to monotonically decrease from upstream to downstream) results in minimum values that are above the minimum value obtained in the limit of Δl tending to zero. The second property is significant with respect to our development in that it permits some flexibility in the construction of objective functions. For example, rather than adopting an objective function whose formulation entails taking the absolute value of a quantity, e.g., Eq. (6.18), a convex function can be formulated in terms of a sum of absolute values of quantities, or similarly, in terms of a sum of squares of quantities. In each of these alternative formulations the individual terms being summed are themselves convex functions.

The general criterion for constrained optimization of a discrete temperature field T_p may be expressed by the relation

$$|Z - Z_m| < \varepsilon_m, \tag{6.C1}$$

where the quantities Z_m and ε_m have values which depend on the type of function adopted for Z. The quantities Z_m and em represent bounds on the level of optimization for a given discrete parameter system in principle and in practice, respectively. That is, the quantity Z_m represents a theoretical limit on optimization of a set of parameters that is due to the inherent unavailability of information concerning a system and the quantity ε_m represents the level of tolerance assumed for minimization of Z.

Appendix D

Semi-Quantitative Proof of Validity of Elliptic-Equation Representation of Temperature Fields Defined within Subdomains Having Constrained Surfaces

Our proof of the validity of an elliptic equation representation of the temperature field within a given subdomain is in terms of both continuous functions and operators and of their discrete analogs consisting of discrete field quantities. Given a temperature field T which satisfies the Laplace equation

Eq. (6.33) within a given subdomain whose boundary is a closed surface S, it follows from Green's theorem[8] that

$$T(x) = -\frac{1}{4\pi}\int_S T(x')\nabla' G(x, x') \cdot n' da' \tag{6.D1}$$

if there is a Green's function $G(x, x')$ such that

$$\nabla'^2 G(x, x') = -4\pi\delta(x - x') \tag{6.D2}$$

and

$$G(x, x') = 0 \tag{6.D3}$$

for positions x within the subdomain and positions x' on the surface S bounding this domain. Next, we note some properties of $G(x, x')$ which follow from Eq. (6.D2). These are

$$\int_S \nabla' G(x, x') \cdot n da' = -4\pi \tag{6.D4}$$

and that $G(x, x')$ is of the general form

$$G(x, x') = \frac{1}{|x - x'|} + F(x, x') \tag{6.D5}$$

where the function $F(x, x')$ satisfies the Laplace equation

$$\nabla'^2 F(x, x') = 0. \tag{6.D6}$$

Equations (6.D1) through (6.D6) provide the general framework for an integral representation of the solution of Eq. (6.33) over a closed domain. Accordingly, construction of a surface integral of the form defined by Eq. (6.D1) and of an associated Green's function whose properties are consistent with Eq. (6.D2) through Eq. (6.D6) implies that the temperature field is represented by Eq. (6.33). Next, we note that the weighted sum defined by Eq. (6.36) can be interpreted as a discrete analog of Eq. (6.D1). That is,

$$-\frac{1}{4\pi}\int_S T(x')\nabla' G(x, x') \cdot n' da' = \sum_{i=1}^{6} W_{E,i} T_{E,i}. \tag{6.D7}$$

The correspondence represented by Eq. (6.D7) establishes our proof.

7

Neural Network Systems Techniques in Weld Modeling and Control

7.1 Introduction
7.2 Welding and Bonding
7.3 Control System Requirements
7.4 Arc Welding as a Multivariable Process
7.5 Traditional Approaches to Arc Welding Modeling and Control
7.6 Neural Networks
7.7 Applications of Neural Network Systems in Weld Modeling and Control
Equipment Setpoint Selection Using the Back Propagation Network • Real-Time Monitoring of Weld Joint Penetration • Dynamic Representation of Resistance Spot Welding Process Input/Output Relationships • Closed-Loop Control of GTA Weld Bead Geometry • VPPA Weld Modeling and Control
7.8 Conclusion

George E. Cook
Vanderbilt University

Robert Joel Barnett
Vanderbilt University

Daniel A. Hartman
Los Alamos National Laboratory

Alvin M. Strauss
Vanderbilt University

7.1 Introduction

With the increasing demands for high quality welds, the need to eliminate the human operator from the feedback path for correction of the welding process has become evident. Of the several manufacturing technologies, welding alone has defied true automation. Even with excellent controllers for the several controllable variables such as voltage, current, and travel speed, the human operator still provides the only commercially-viable feedback loop. Recent research in intelligent, closed-loop, automatic control of the welding process is making steps in eliminating the human operator from the feedback loop while providing high quality consistent welds. However, it is still necessary to have the human operator monitor the weld in progress and make corrections to compensate for minor variations in joint fit-up, plate thickness, variations in thermal conditions, or one or more of several of the uncontrollable variables.

A review of some of the more noteworthy intelligent control techniques for automated welding follows. This is by no means an exhaustive review of artificial neural network applications for intelligent weld control. Furthermore, other equally capable intelligent control schemes using fuzzy logic, neuro-fuzzy models, and rule-based expert systems have been successfully applied to intelligent weld control.

Unfortunately, an all-inclusive intelligent weld control review is beyond the scope of this chapter. Nevertheless, numerous references at the end of this chapter are included for further investigation.

7.2 Welding and Bonding

Most welding processes require the application of heat or pressure, or both, to produce a bond between the parts being joined. The welding control system must include means for controlling the applied heat, pressure, and filler material, if used, to achieve the desired weld microstructure and mechanical properties.

Welding usually involves the application or development of localized heat near the intended joint. Welding processes that use an electric arc are the most widely used in the industry. Other externally applied heat sources of importance include electron beams, lasers, and exothermic reactions (oxyfuel gas and thermit). For fusion welding processes, a high energy density heat source is normally applied to the prepared edges or surfaces of the members to be joined and is moved along the path of the intended joint. The power and energy density of the heat source must be sufficient to accomplish local melting.

7.3 Control System Requirements

Insight into the control system requirements of the different welding processes can be obtained by consideration of the power density of the heat source, interaction time of the heat source on the material, and effective spot size of the heat source.

A heat source power density of approximately 10^3 W/cm^2 is required to melt most metals [18]. Below this power density the solid metal can be expected to conduct away the heat as fast as it is being introduced. On the other hand, a heat source power density of 10^6 or 10^7 W/cm^2 will cause vaporization of most metals within a few microseconds, so for higher power densities no fusion welding can occur. Thus, it can be concluded that the heat sources for all fusion welding processes lie between approximately 10^3 and 10^6 W/cm^2 heat intensity. Examples of welding processes that are characteristic of the low end of this range include oxyacetylene welding, electroslag welding, and thermit welding. The high end of the power density range of welding is occupied by laser beam welding and electron beam welding. The midrange of heat source power densities is filled by the various arc welding processes.

For pulsed welding, the interaction time of the heat source on the material is determined by the pulse duration, whereas for continuous welding the interaction time is proportional to the spot diameter divided by the travel speed. The minimum interaction time required to produce melting can be estimated from the relation for a planar heat source given by [18]

$$t_m = [K/p_d]^2 \tag{7.1}$$

where p_d is the heat source density (watts per square centimeter) and K is a function of thermal conductivity and thermal diffusivity of the material. For steel, Eagar gives K equal to 5000 W/cm^2/s. Using this value for K, one sees that the minimum interaction time to produce melting for the low power density processes, such as oxyacetylene welding with a power density on the order of 10^3 W/cm^2, is 25 s, while for the high energy density beam processes, such as laser beam welding with a power density on the order of 10^6 W/cm^2, is 25 μs. Interaction times for arc welding processes lie somewhere between these extremes.

Examples of practical process parameters for a continuous gas tungsten arc weld (GTAW) are 100 A, 12 V, and travel speed 10 ipm (4.2 mm/s). The peak power density of a 100-A, 12-V gas tungsten arc with argon shielding gas, 2.4-mm diameter electrode, and 50-degree tip angle has been found to be approximately 8×10^3 W/cm^2. Assuming an estimated spot diameter of 4 mm, the interaction time (taken here as the spot diameter divided by the travel speed) is 0.95 s. At the other extreme, 0.2-mm (0.008-in.) material has been laser welded at 3000 in./min (1270 mm/s) at 6 kW average power. Assuming a spot diameter of 0.5 mm, the interaction time is 3.94×10^{-4}s.

Spot diameters for the high density processes vary typically from 0.2 mm to 1 mm, while the spot diameters for arc welding processes vary from roughly 3 mm to 10 mm or more. Assuming 1/10 the spot diameter for positioning accuracy, we conclude that a typical positioning accuracy requirement for the high power density processes is on the order of 0.1 mm and for arc welding processes is on the order of 1 mm. The required control system response time should be on the order of the interaction time, and hence, may vary from seconds to microseconds, depending on the process chosen. With these requirements it can be concluded that the required accuracy and response speed of control systems designed for welding increases as the power density of the process increases. Furthermore, it is clear that the high power density processes must be automated because of the human operator's inability to react quickly and accurately enough.

7.4 Arc Welding as a Multivariable Process

In this section the Gas Tungsten Arc Welding (GTAW) process (see Fig. 7.1), sometimes referred to as the Tungsten Inert Gas (TIG) process, is discussed as an example for artificial neural network applications. In this process the electric arc is sustained between a nonconsumable tungsten electrode and the workpiece. The electrode is usually at a negative voltage potential with respect to the workpiece and current is, therefore, carried across the arc gap through ionized plasma. The power source providing the electrical current is operated in a current-control mode. While the arc current is controlled by the power source, the voltage across the arc is largely determined by the length of the arc gap. An inert gas, such as Argon or Helium, is routed along the electrode down through the gas cup such that it encompasses the molten weld pool as well as the arc. The gas serves to provide the medium for ionization and plasma formation, and to shield the weld pool from the atmosphere and thus prevent oxidation of the molten metal. Filler wire is usually fed into the pool from the front end of the moving torch, where it melts and combines with the molten base metal of the workpiece.

Any arc welding process can be controlled by a number of parameters, and the ultimate objectives of the process are specified in terms of numerous parameters as well (refer to Fig. 7.2). As a result, any arc welding process can generally be viewed as a multiple-input/multiple-output system. The lack of reliable, general, and yet computationally fast, physical models of this multivariable system makes the design of

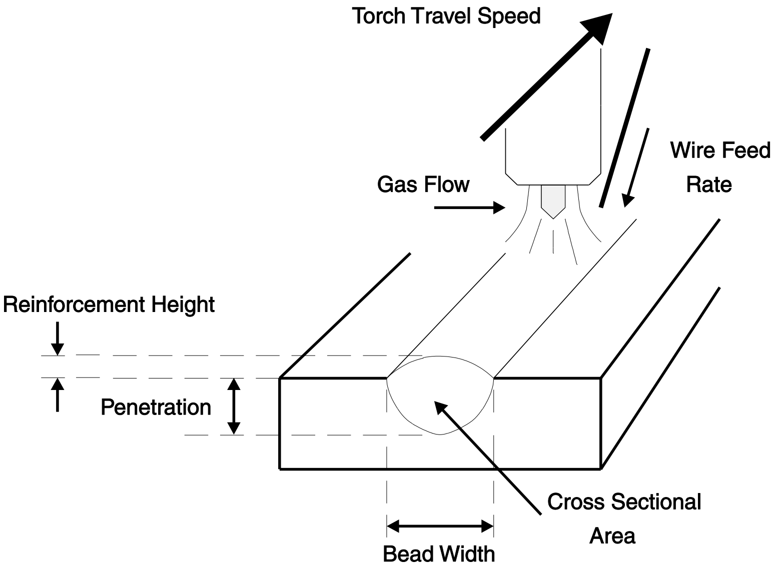


FIGURE 7.1 GTA weld process parameters.

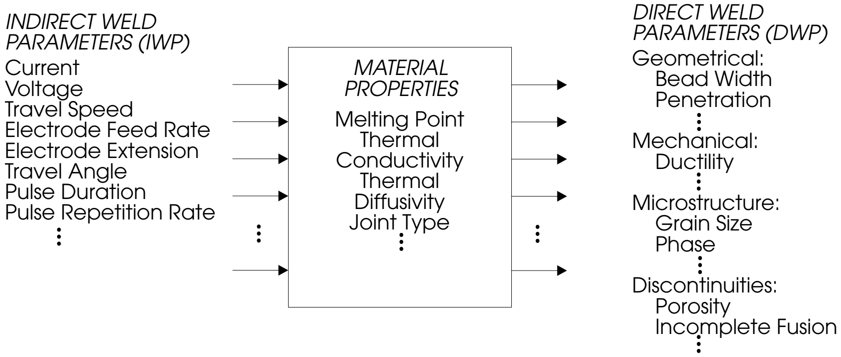


FIGURE 7.2 Input and output variables of the general arc welding process.

a generalized real-time controller for arc welding nontrivial. Each process has its own characteristics which usually are related to a number of external parameters. Generally, these relationships are not well quantified.

One of the basic issues to be considered regarding the generalized control for arc welding is to determine what is to be controlled and which parameters are accessible to enact control actions on the process. In other words, the input parameters and the output parameters of the process, or the plant, have to be determined. The input parameters to any arc welding process typically include those controlled by the welding equipment, such as arc current, voltage, torch travel speed, and filler wire rate. Most of these equipment parameters may be adjusted on-line and thus they are available for real-time control of the welding process. Other input parameters may be selected by the welder, but they are usually static and not changeable during welding. These parameters include the electrode type and dimensions, filler wire composition and dimensions, workpiece dimensions and composition, etc.

7.5 Traditional Approaches to Arc Welding Modeling and Control

Applications of control for arc welding have thus far been largely focused on the control of one, or at most two, parameters of the process. The control techniques for these parameters have usually been limited to classical feedback control, based on the assumptions of linear and time invariant behavior of the controlled systems. Examples of such control schemes include voltage and current control of the welding power source, control of the arc voltage through adjustment of the arc length, and other isolated single-variable control applications. The welding control applications mentioned above are usually adequate for their specific purposes, i.e., to maintain one or two system parameters at predefined values. These traditional schemes, however, are not easily applicable to the more complex task of directly controlling outputs of the entire process through the available equipment parameters, i.e., controlling the DWP in terms of the IWP [1, 3]. Conceptually, the IWPs can be thought of as causes, with the DWPs as effects, both being related by the material properties of the weldment itself.

One of the main obstacles to designing a general welding controller for any arc welding process, such as GTAW, is that the process to be controlled is not thoroughly understood. The current approaches to welding rely extensively on human expertise and trial-and-error methods. For a given task, the human welder selects the process constants (electrode size, material and tip preparation, gas type and flow rate, etc.) and sets the variable equipment parameters (arc current, voltage, torch travel speed, filler wire rate, etc.). These selections are usually based on past experience. Through initial trials the variable equipment parameters are adjusted, and this process is repeated until the desired results (weld bead size, appearance, etc.) are obtained. Improvements on two aspects of this procedure can be proposed at this point. First, improvements in the initial selection of the indirect weld parameters would reduce the number of iterations necessary to obtain the desired direct weld parameters. This initial selection is static and can be carried out off-line, prior to welding. Secondly, once the process has started, sensory feedback and other available process information can be monitored in real-time and used to update the IWP which control the welding equipment. These updates must be obtained on-line and the corresponding control actions must be executed at a high enough rate to effectively control short-term transients of the process.

Controlling a multivariable system is not a trivial task without an adequate quantitative model. In the case of arc welding, relationships between the various process inputs and outputs are not well defined. Furthermore, the process variables are coupled (i.e., any given input parameter affects more than one output parameter) and in general the welding processes are nonlinear (the output parameters are not adequately described in terms of a linear combination of the input parameters and their time derivatives). All of these facts add to the difficulty of designing a general controller for arc welding. Substantial research has been carried out to mathematically model and control the GTAW process, as well as other arc welding processes. Each welding process is primarily governed by specific control variables, as discussed in [12]. The available weld process models can be classified from either of two viewpoints. One classification arises from the methods by which the models are derived, while the other is concerned with the ability of the models to describe the dynamic behavior of the process as well as the static one.

In broad terms, weld process models are either derived from the fundamental physics of heat transfer, or they are constructed from empirical data. The models derived from heat transfer physics frequently assume that the arc can be modeled as a heat source of a given form (a point source, a disk-shaped source of heat, etc.) and then the laws of heat conduction are applied to calculate the temperatures at various points in the workpiece [65, 66, 69]. Of specific interest is the boundary at which the temperature equals the melting temperature of the welded metal, i.e., the boundary of the molten weld pool. These results may either be derived analytically or computationally to a varying degree. The analytical derivations usually require significant simplifications, such as assuming that the welded workpiece is either infinitely thick or of a specific shape, the thermal properties of the molten pool (e.g., heat conduction and density) are homogeneous and the same as those of the solid metal, etc. As a result the analytically derived models are usually fairly inaccurate when compared to the results of actual welding operations. On the other hand they frequently offer some insight into the mechanisms of the welding process and they may illustrate how the individual process inputs and outputs are related.

Computer solutions based on the physics of heat conduction can be made more accurate by including some of the calculations omitted in the analytical simplifications. Finite-difference techniques and related methods carry the computational solution of the heat equation to its fullest accuracy [21, 22, 20]. The drawbacks of such approaches, however, are that they are computationally intensive (which usually renders them impractical for real-time use) and they do not offer the insight into the interactions of process variables that the analytical models may give.

An alternative to the physics-based models, discussed above, are the empirically derived models [13, 14]. These models may simply consist of one or more equations relating the process outputs to the process inputs, and derived by obtaining a best fit of experimental data to the given equations. In such cases the models are derived without any consideration of the underlying physics of the process. A number of weld process models can be placed between the two extremes of pure physics-derived and empirical models. Frequently, physics-based models are derived using the necessary approximations and then various "empirical constants" and other unspecified variables are tuned until the model adequately agrees with experimental welding data. By the same token, empirical models frequently use pieces of general knowledge of the process, derived from the underlying physics. Therefore, the physics-based and the empirical models define the opposite extremes in approaches to weld modeling.

For the purpose of system control it is important to distinguish between static models and dynamic ones. A static model describes the process in its steady-state, i.e., it does not consider short-term transients of the process. A dynamic model, however, yields both the dynamic and the static responses of the process, and thus it is more general. The lack of dynamic considerations is usually tolerable when the initial setpoint values for the equipment parameters of the process are selected with the aid of a model. Once the process has started, however, its dynamics must be adequately incorporated into the control scheme. Some adaptive, dynamic control schemes for the GTAW process are discussed in [50].

There are at least two roles for models in a generic nonadaptive weld process controller. First, a model can be used in defining the initial equipment parameters of the process. The welder specifies the desired DWP, such as weld bead width, penetration, etc., and the model can be used to arrive at suitable IWP, such as welding current, travel speed, etc. Secondly, a model can be executed in parallel with the actual process and provide calculations for DWP that cannot be measured directly in real-time. Thus, a weld model can provide the controller at any time with an estimate of the weld bead penetration, although it may not be measurable in real-time. Adaptive controllers may require a process model to compare against the physical process for computation of errors used to inact parameter adaptation.

7.6 Neural Networks

Recent successes in employing artificial neural network models for solving various computationally difficult problems have inspired renewed research in the area. Early work by McCulloch [55] and Widrow [56] focused largely on mathematical modeling, while more recent research has augmented theoretical analysis with computer simulations and implementation demonstrations. Numerous variants of pattern classifiers using neural networks have been studied by Hopfield [60] and Lippmann [59]. Introductory texts on the subject may be found in [57] and [58].

As the name indicates, a neural network resembles, to a certain degree, biological nervous systems as we currently understand them. While most traditional computers rely on a single or few computational units to perform their tasks, the neural network typically consists of a relatively large number of computational units, connected with an even larger number of communication links. The underlying principle aims to examine numerous hypotheses simultaneously and process data in a distributed fashion. In essence the neural network is a self-adaptive structure which incrementally alters its inner workings until it achieves the desired performance.

The fundamental building block of an artificial neural network is the perceptron which was introduced by Rosenblatt in [54]. Originally designed for pattern recognition and as a research tool for modeling brainlike functions of a biological system, the perceptron pattern-mapping architecture can be generalized to the processing element present in the back-error propagation systems used today. These processing elements (as shown in Fig. 7.3) are linked together with variable weights into a massively parallel environment. A neural net can achieve human-like performance due to its ability to form competing hypotheses simultaneously.

A neural network and its adaptation procedure using back propagation is best illustrated by an example. Figure 7.4 shows a small neural network consisting of eight nodes, arranged in two hidden layers of three nodes each and one output layer of two nodes. Each node i in the first hidden layer produces a single numeric output which we denote as $x_i^{(1)}$. Similarly the nodes of the second hidden layer are labeled $x_0^{(2)}$, $x_1^{(2)}$, and $x_2^{(2)}$. The three inputs and two outputs of the network are x_0 through x_2 and y_0 through y_1 respectively. Each node accepts numeric data through a number of input links, each of which multiplies

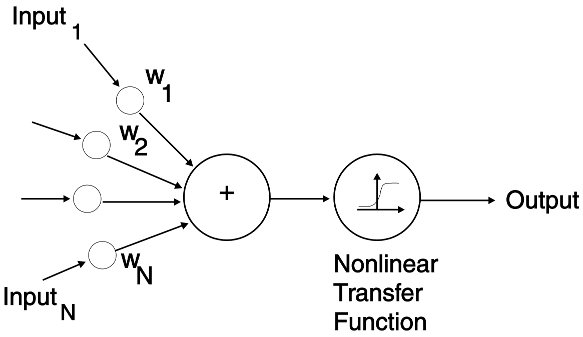


FIGURE 7.3 The processing element of a neural network.

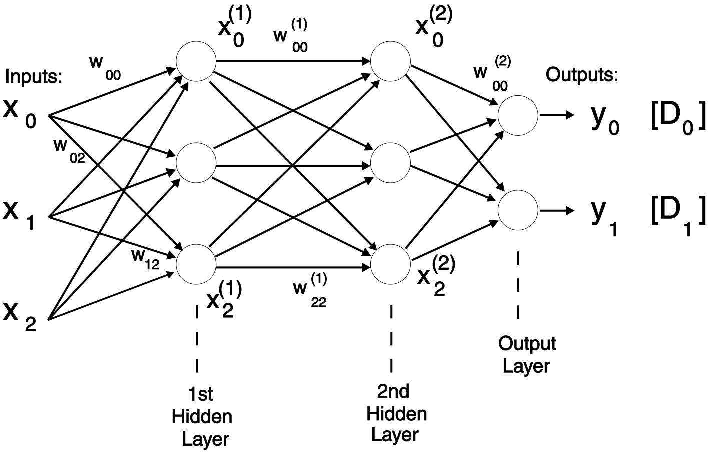


FIGURE 7.4 Three-input, two output neural network, using two hidden layers of three nodes each, fully interconnected.

the input data with a weight factor. The weight factor associated with the link from $x_i^{(1)}$ to the node producing $x_j^{(2)}$ is annotated as $w_{ij}^{(1)}$ and a similar convention holds for the links between other layers. Each node calculates its output by summing its weighted inputs and using the result s as the argument of a nonlinear function associated with the node. For our application this function is the same for all nodes:

$$f(s) = [1 + \exp[-(s - c)]]^{-1} \tag{7.2}$$

where s is the sum of the node inputs and c is an internal offset value. Clearly the node output will be confined to the range $0 < f(s) < 1$. Because the limiting values, 0 and 1, will only be approached as s approaches $+/-$ infinity, all input and output data are scaled so that they are confined to a subinterval of $[0 \ldots 1]$. A practical region for the data is chosen to be $[0.1 \ldots 0.9]$. In this case each input or output parameter p is normalized as p_n before being applied to the neural network according to:

$$p_n = \left[\frac{(0.9 - 0.1)}{(p_{\max} - p_{\min})}\right](p - p_{\min}) + 0.1 \tag{7.3}$$

where $p_{\max}$ and $p_{\min}$ are the maximum and minimum values, respectively, of data parameter p. The network starts calculating its output values by passing the weighted inputs to the nodes in the first layer. The resulting node outputs of that layer are passed on, through a new set of weights, to the second layer, and so on until the nodes of the output layer compute the final outputs.

Before practical application, the network has to be trained to perform the mapping of the three input parameters to the two output parameters. This is done by repeatedly applying training data to its inputs, calculating the corresponding outputs by the network, comparing them to the desired outputs, and altering the internal parameters of the network for the next round. The training starts by assigning small random values to all weights (w_{ij}) and node offsets (c_j) in the network. The first three input data values are presented to the network which in turn calculates the two output values. Because the initial weights and node offsets are random these values will generally be quite different from the desired output values, D_0 and D_1. Therefore, the differences between the desired and calculated outputs have to be utilized to dictate improved network values, tuning each weight and offset parameter through back propagation. The weights preceding each output node are updated according to

$$w_{ij}(t + 1) = w_{ij}(t) + nd_j x_i^{(2)} \tag{7.4}$$

where n is a correction gain and d_j is the correction factor

$$d_j = y_j(1 - y_j)(d_j - y_j) \tag{7.5}$$

Clearly, each weight will be increased if the calculated output from its node is less than the desired value, and vice versa. The correction factors used to update weights preceding the hidden layer nodes are updated according to

$$d_j = x_j(1 - x_j)\sum_k (d_k - w_{jk}) \tag{7.6}$$

where the k applies to the node layer succeeding the one currently being updated. The offset parameter c of each node is treated as an additional weight factor and updated in the same manner.

The weights and offsets of the neural network are recalculated during the back propagation as outlined above. Then the network repeats the calculation of the output values based on the same input data, compares them to the desired output values, and readjusts the network parameters through yet another back propagation phase. This cycle is repeated until the calculated outputs have converged sufficiently close to the desired outputs or an iteration limit has been reached. Once the neural network has been tuned to the first set of input/output data, additional data sets can be used for further training in the same way. To ensure concurrent network adaptation to all sets of data, the entire training process may be repeated until all transformations are adequately modeled by the network. This requires, of course, that all the data sets were obtained from the same process and therefore the underlying input/output transformation is consistent.

As noted above, the training iteration process may be terminated either by a convergence limit or simply by limiting the total number of iterations. In the former case we use an error measure e defined as the following:

$$e = \max_{k=1\ldots K}\left[\sum_{m=0}^{M-1} (d_{k,m} - y_{k,m})^2\right] \tag{7.7}$$

where K is the number of input/output data sets used for training, M is the number of network output parameters in each data set, and $(d_{k,m} - y_{k,m})$ is the error in the network calculation of parameter m in data set k. The error measure, e, changes after each round of network weight adjustments. In the long run e decreases as the network is refined by training iterations. Using this indicator one can program the network to terminate the iterative tuning process as soon as e reaches some threshold value, e_0. Alternatively, a given network may not be able to reduce the error measure down to the specified e_0. In that case the iterations may be terminated by simply specifying a maximum number for them.

The training mode, as described above, is a precondition for actually applying the neural network in the application mode. In this mode entirely new input data is presented to the network which, in turn, predicts new outputs based on the transfer characteristics learned during the training. If this new data is obtained from the same local region of operation of the process as during the training phase, data from the input/output relations should be governed by the same underlying process and the neural network should perform adequately. The neural network is not updated in the application mode.

When compared to other modeling methodologies, neural networks have certain drawbacks as well as advantages. The most notable drawback is the lack of comprehension of the physics of the process. Relating the qualitative effects of the network structure or parameters to the process parameters is usually impossible. On the other hand, most physical models resort to substantial simplifications of the process and therefore trade accuracy for comprehensibility. The advantages of neural models include relative accuracy, as illustrated in the following sections, and generality. If the training data for a neural network is general enough, spanning the entire ranges of process parameters, the resulting model will capture the complexion of the process, including nonlinearities and parameter cross couplings, over the same ranges. Model development is much simpler than for most other models. Instead of theoretical analysis and development for a new model the neural network tailors itself to the training data. Finally, the neural network can calculate its result relatively fast, as the input data is propagated once through the network in the application mode.

7.7 Applications of Neural Network Systems in Weld Modeling and Control

Equipment Setpoint Selection Using the Back Propagation Network

Selecting the appropriate indirect weld parameters for a given weld geometry is a nontrivial task. Generally, each parameter of the welding equipment affects a number of geometrical attributes of the weld pool. Current, for instance, affects both weld pool width and penetration. The human welder or weld designer usually selects the indirect weld parameters based on previous experience, handbook recommendations, etc., and then fine tunes the selection with trial experiments. This approach can be complemented or replaced by the neural network scheme.

Zeng et al. demonstrate that an ANN can provide equipment welding parameters necessary to produce high-integrity welds [44]. In their research they utilized a commercial neural network package called Brainmaker [42] to simulate a manual TIG welding parameter predictor. The IWP chosen for their neural predictor are welding current and voltage, the amount of filler rod consumed, weight of weld metal deposited, and arcing time per meter of weld. The inputs to the network are material type, joint type, welding position, shielding gas, sheet thickness, joint gap, and filler rod diameter. Material type and joint type are represented as symbolic variables such that several input neurons, allowing only a binary input value (i.e., 0 or 1), facilitate the representation of different materials (e.g., mild steel, stainless steel, etc.) and different welding positions (e.g., flat position, H/V position, vertical position, or overhead position).

Their network was trained using the back propagation learning algorithm. The raw data was provided by the Welding Institute in which 40 input/output training instances were used. Material types varying between mild steel and stainless steel, and each of the four possible welding positions were included in the training set. Extensive experiments highlighted that only one hidden layer did not provide the error tolerance desired by Zeng. Therefore, two hidden layers of 50 neurons in the first layer and 25 neurons in the second layer converged to an acceptable error tolerance of 4%. Tests indicated that good prediction was achieved.

The back propagation network has also been implemented by Andersen, et al. [6, 7] to relate welding process variables to weld bead geometry. They employed two neural networks. One network was trained to derive DWP's from IWP's while another network was trained to yield IWP's from DWP's. This approach is illustrated in Fig. 7.5. The user specifies the required weld geometry by the reference or setpoint DWP's (DWP_{REF}). Assuming that the network has been properly trained off-line with previous sets of actual weld parameters, the network produces a set of IWP's that can be used as weld equipment settings.

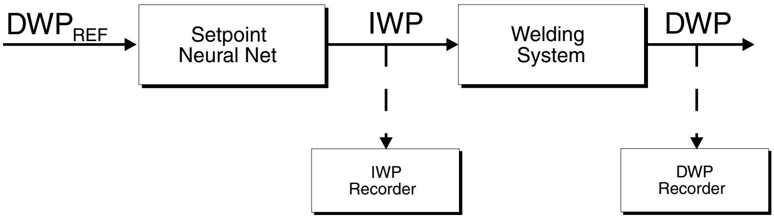


FIGURE 7.5 Indirect weld parameter selector using neural network.

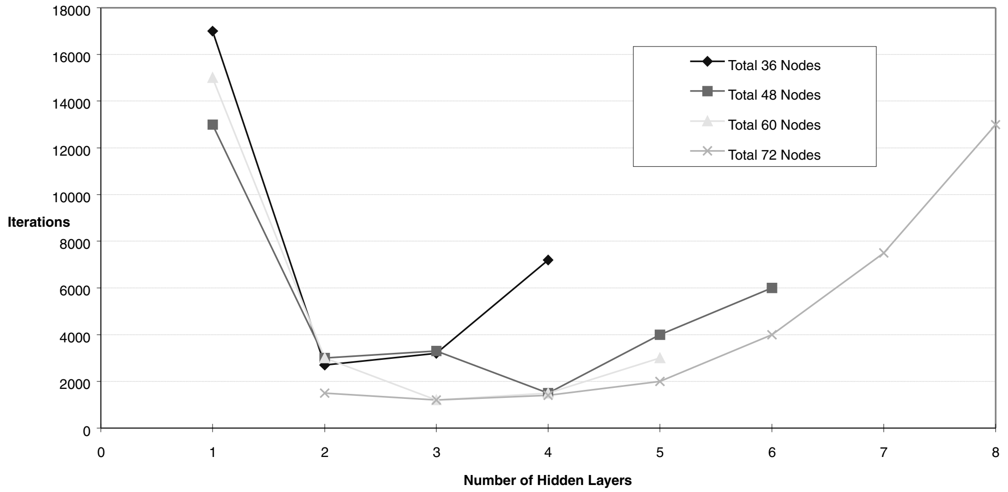


FIGURE 7.6 Number of iterations required for convergence of various network configurations. The error threshold e_0 is 0.09 and the correction gain η is 0.1 for all networks.

If the network parameters include the ones having principal roles in controlling the weld pool, the welder can achieve the required DWP's with minimal experimentations or experience. While the weld is performed, the IWPs may be recorded in the IWP recorder, and similarly, the resulting DWPs can be determined and stored in the DWP recorder after welding. Once these data are available the neural network can be refined by additional off-line training with the new data, and thus its characteristics are continuously updated as future welds are produced.

Andersen used actual GTA weld data to evaluate the accuracy of the neural networks for weld modeling. The data consisted of values for voltage, current, travel speed, and wire feed speed, and the corresponding bead width, penetration, reinforcement height, and bead cross-sectional area. In all, 42 such data sets were used, of which 31 data sets were selected at random and used for training purposes while the remaining 11 data sets were presented to the trained networks as new application data for evaluation purposes. Thus the networks were evaluated using data that had not been used for training.

Networks consisting of 36, 48, 60, and 72 nodes using various number of hidden layers were built. The 31 sets of welding data were used for training each of the networks. Using a learning rate of η = 0.1, the number of training iterations required to reduce the conversion measurement, e_0, down to 0.09 is illustrated in Fig. 7.6. Andersen defined a training iteration as one round of network adaptation to all sets of training data. Figure 7.6 clearly indicates that there is an optimum number of hidden layers that requires the least amount of training iterations for this particular problem. Although the training iterations take place off-line and are therefore irrelevant to on-line modeling, they can be fairly time consuming. Thus, it is clear that training time optimization is advantageous for extensive experiments.

Each network configuration resulted in comparable performance. Figure 7.7 summarizes the performances of each of the networks for the four DWPs. The bars show standard deviations of the modeling errors for the various networks using the 11 tested data sets. Most of the errors are on the order of 5 to 10%, with penetration errors typically in the 20% range. These results agree with other similar experiments reported by Andersen, in that modeling accuracy is typically on the order of 10 to 20%.

An observation relating to the weld modeling experiments should be noted here. The precision of the bead measurements was 0.1 mm, which corresponds to 2 to 7% precision for the average bead width and penetration, respectively. Furthermore, inaccuracies in measurements of the data, which were used to train the neural network model, tend to degrade the general performance of the model. Width measurements are generally more reliable than penetration measurements, as they are made in several locations along the top of the bead. A penetration measurement is usually made on a single cross section,

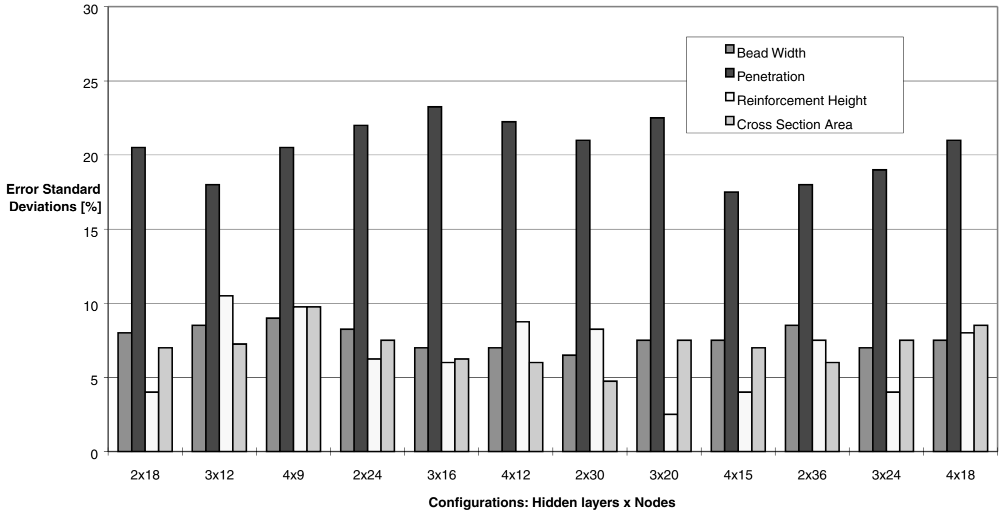


FIGURE 7.7 Standard deviations of mapping errors (%) for various network configurations, shown for all four DWPs.

as it requires chemical etching, which results in a relatively blurred boundary between the bead and the surrounding base metal. This difference is reflected in the consistently lower accuracy of the penetration modeling, compared with the width modeling.

The two neural network schemes presented by Zeng and Andersen apply to static parameter settings only. Therefore, dynamic control of the welding process is left to the various controllers of the welding equipment, such as the automatic voltage controller (AVC), power source (usually constant current for GTAW), etc. Thus, it is the responsibility of these controllers to maintain the parameters, selected by the neural network, constant in the presence of noise or process perturbations.

Real-Time Monitoring of Weld Joint Penetration

A rather elaborate implementation of monitoring weld joint penetration using machine vision and a high-shutter-speed camera assisted with pulsed laser illumination is presented by Kovacevic, et al. [35, 36, 37, 38, 39]. In their research, they employ an ANN to emulate the human operator's ability to extract weld penetration information from the geometrical appearance of the weld pool [36]. Conceptually, Kovecevic believes that the human welder relies on the geometrical appearance parameters, which include size and shape information, of the weld pool to represent weld penetration. Kovacevic's model uses an empirically derived relationship between the sag of the weld near the rear of the pool and the root face bead width [15, 29, 30]. Thus, their ANN weld penetration estimator is trained with the length and the included rear angles of the weld pool to represent the pool's size and shape, respectively.

The objective of Kovacevic's experiments was to monitor, in real-time, the full penetration state of the weld pool which is described by the root face bead (backside) width. Extensive neural network configurations were tested in order to examine the possibility of using different pool parameters as inputs to represent the weld penetration. Each of the networks, however, contained only one output node which was responsible for calculating the backside bead width.

They employed a commercial neural network software package to facilitate the training of the various neural network configurations. An extended delta-bar-delta (EDD) learning algorithm was used to overcome the slow convergence rate associated with the back propagation learning algorithm. Only one hidden layer was used in which the number of nodes, n_2, was calculated by the following equation:

$$\frac{N}{10(n_1 + n_3)} < n_2 < \frac{N}{5(n_1 + n_3)} \tag{7.8}$$

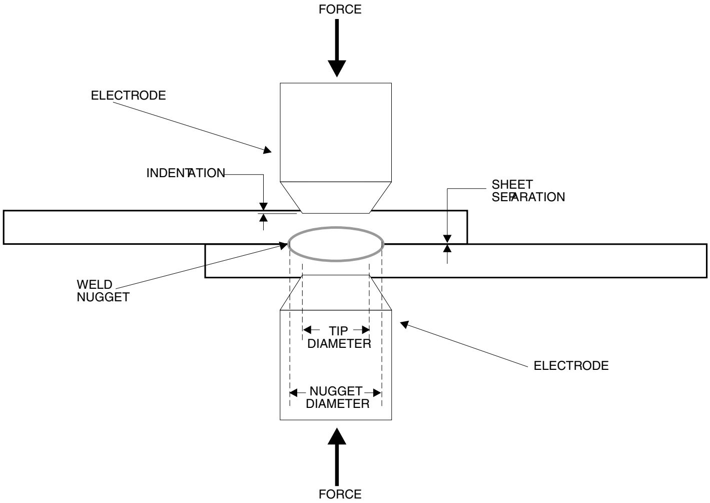


FIGURE 7.8 Resistance spot welding process.

where N is the number of samples, and n_1 and n_3 are the number of elements in the input and output layer, respectively.

Kovacevic found that while none of the input parameters (width, length, length-to-width ratio, or rear angles) alone can accurately assess the penetration, a combination of the length and rear angles of the weld pool provided sufficient information to accurately model the relationship between the weld pool and weld penetration. Furthermore, monitoring was possible in real-time at a rate of 5 Hz.

Dynamic Representation of Resistance Spot Welding Process Input/Output Relationships

Used typically for joining one sheet-metal panel to another, spot welding, also known as resistance welding or resistance spot welding (RSW), is one of the oldest production welding methods. The resistance welding process differs from the arc welding processes in that filler wire is rarely used and fluxes are not employed. As depicted in Fig. 7.8, resistance welding uses pressure and electrical resistance through two pieces of metal to form a molten nugget at the faying surface between the two sheets. Weld quality is affected by the following factors: current applied, on-and-off cycle timing, electrode pressure, and the shape and condition of the electrode.

Messler, et al. use a neural network and fuzzy logic controller to adjust the amount of heat input in order to compensate for variations and errors in the RSW process [27]. Their intelligent control system relies on the electrode displacement, caused by thermal expansion between the electrodes during nugget formation, as the single parameter input to indicate weld nugget growth. Desired electrode displacement vs. percent heat input and welding time is calculated by a neural network. A fuzzy logic controller compares the desired electrode displacement with the actual electrode displacement and then appropriately adjusts the power delivered in real-time.

Messler states that there are numerous sources of variations and errors which include: material composition and coatings, workpiece stack-up thickness, fit-up, surface roughness, and cleanliness; electrode deformation, wear, or contamination; welding machine cooling efficiency, force consistency, and mechanical compliance; and electrical dips or surges. Nevertheless, numerous mathematical models have been developed to represent the ideal electrode displacement curve [70]. Unfortunately, these mathematical models have limitations in a practical application due to the previously mentioned sources of variations and errors. Furthermore, a comprehensive analysis of the RSW process is extremely complex due to the interactions between electrical, thermal, mechanical, and metallurgical phenomenon of the process.

Consequently, Messler uses an ANN to capture the dynamics of the RSW process through the training of actual welding data.

A time-delay neural network was developed by Messler in order to map the percent heat input to electrode displacement. In addition to percent heat input, the time-dependent electrode displacement output of the network is fed back as fixed-time delayed inputs. These tapped delay lines are necessary in order to consider the dynamic electrode displacement response. The result is a single-channel temporal sequence neural network which can map the percent heat input to electrode displacement.

Four different neural network structures using various time delays were trained using the back propagation learning algorithm in order to maximize performance. Training data sets were obtained under normal production welding conditions, i.e., no special surface preparation or cleaning was performed and electrodes were replaced as needed, for various heat inputs. A second training set was evaluated when a worn electrode was used under similar heat inputs. In both cases the lowest maximum error of 5 % was obtained using four input nodes. (percent heat input and three time delays), seven hidden layer nodes, and one output node.

Closed-Loop Control of GTA Weld Bead Geometry

An approach to closed-loop control, using neural networks, has been attempted by a number of researchers using various welding processes, such as GTAW, GMAW, RSW, VPPAW, etc., and a wide range of sensing devices, such as optical, arc, infrared, acoustical, ultrasonic, etc. [5, 6, 7, 11, 25, 27].

Andersen developed a digital welding control system for the GTAW process in which the objective was to maintain constant weld bead dimensions when a variation in the welding environment was encountered, such as a variation in the thickness of the plate upon which the weld was being performed [6]. This is a practical consideration when welding items of relatively complex shapes, where the varying thermal conditions can affect the geometry of the molten pool.

Andersen's control system is most readily explained by reference to Fig. 7.9, which illustrates the main components. The desired bead width and penetration are specified by the user as W_{REF} and P_{REF}, respectively. These parameters, as well as the workpiece thickness, H, are fed to a neural network setpoint selector, which yields the nominal travel speed, current, and arc length (v_0, I_0, and L_0, respectively). Arc initiation and stabilization is controlled in an open-loop fashion by the weld start sequencer. Given the desired equipment parameters the arc is typically initiated and established at a relatively low current, with the other equipments parameters set at some nominal values. Once the arc has been established the equipment

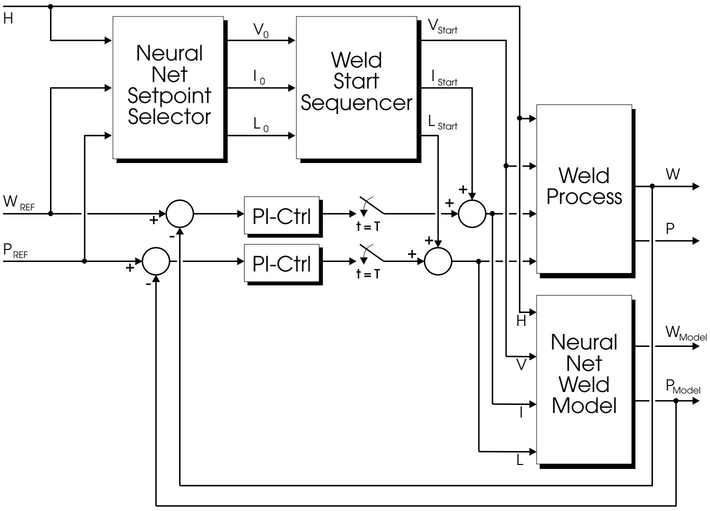


FIGURE 7.9 Closed-loop welding control system.

parameters are ramped to the setpoint values specified by the neural network. When the setpoint values have been reached, at time $t = T$, the closed-loop control is enacted. The bead width from the process can be monitored in real-time. A real-time penetration sensor is not readily available, therefore, a second neural network is run in parallel with the process to yield estimates of the penetration. The measured bead width and the estimated penetration are subtracted from the respective reference values, processed through proportional-plus-integral controllers, and added to the final values obtained from the setpoint sequencer. When a workpiece thickness variation is encountered in the process, the system adjusts the current and the travel speed accordingly to maintain constant bead geometry.

The control loop experiment was run using mild steel for the workpiece material. Plates of two thicknesses, 0.125 in. and 0.250 in., were joined together and an arc using the nominal arc parameters ($I = 100$ A, $L = 0.100$ in., $v = 6$ ipm) was run across the boundary between the plates, from the thicker section to the thinner one. The bead width and penetration were 0.142 in. and 0.035 in., respectively, on the thicker plate. With the controller disabled (and thus the equipment parameters remaining unchanged), the bead width increased to 0.157 in. and the penetration increased to 0.047 in. when the weld pool entered the thinner plate. With the controller enabled, the width and the penetration remained the same on the thin plate as they were on the thick plate, with only a slightly discernible transient.

VPPA Weld Modeling and Control

Artificial neural networks were evaluated for monitoring and control of the variable polarity plasma arc welding process by Cook, et al. in [5]. Three areas of welding application were investigated: weld process modeling, weld process control, and weld bead profile analysis for quality control.

Plasma arc welding (PAW) of aluminum has been extensively used and developed at NASA's Marshall Space Flight Center using square wave ac with variable polarity (VPPA) [67]. With the variable polarity process, oxide removal before welding is not required for most aluminum alloys. NASA and its subcontractors use the VPPAW process in the space shuttle external tank fabrication.

The plasma arc differs from the ordinary unconstricted welding arc in that, while ordinary weld arc jet velocities range from 80 to 150 m/s, plasma arc jet velocities range from 300 to 2000 m/s in welding usage [68]. The high plasma arc jet velocities are produced by heating the plasma gas as it passes through a constricting orifice. In VPPA welding of certain ranges of metal thickness, appropriate combinations of plasma gas flow, arc current, and weld travel speed will produce a relatively small weld pool with a hole (called a keyhole) penetrating completely through the base metal.

VPPAW process modeling was investigated to determine if attributes of the weld bead, such as crown width, root width, or penetration could be predicted based on the corresponding equipment parameters (arc current, torch standoff, travel speed, etc.). Emphasis was placed on direct control of the weld bead geometries, where the user could specify the desired crown width and root widths, and the proposed control system would select and maintain the equipment parameters necessary to achieve the desired results. Various neural network methodologies were investigated for the purpose of arc weld modeling and control.

Additionally, artificial neural networks were used to analyze digitized weld bead profiles, obtained from a laser-based bead surface scanner presently used at NASA. The networks provided an improved method for automatically detecting the location of significant bead attributes, such as crown, undercuts, and edges.

Process Modeling

Data used for the VPPAW process modeling was obtained from the NASA Marshall Space Flight Center [8], which consisted of the following parameters: torch standoff, forward current, reverse current, travel speed, crown width, root width. Of the 13 VPPA welding runs, 10 were used for network training while the other three were used exclusively for testing.

A back propagation neural network was constructed to model the welding data (refer to Fig. 7.10. Several network configurations were tested by varying the number of nodes between 5 and 20 in the

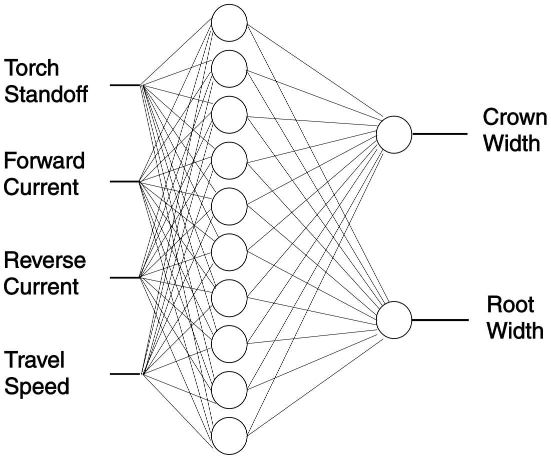


FIGURE 7.10 The neural network model of the VPPAW process.

single hidden layer. The input parameters to the network were the torch standoff, forward current, reverse current, and torch travel speed; while the model outputs were the weld crown width and the root width. The network was trained for 10,000 iterations. Further training did not improve the modeling performance of the network, and it turned out that with excessive training (e.g., through 50,000 iterations and beyond) the network converged to "memorization" of the training data, rather than capturing the generalities of the process. The trend to memorize, rather than learn, when excessive training is applied is a well known characteristic of the back propagation learning algorithm and it was observed in most experiments carried out for this project [9]. The solution to this was simply to observe the general trend of decreasing errors of the testing data and terminate training when the testing data error flattened out or started decreasing.

Only minor performance differences were noticed between the different network configurations. Once the networks had been trained adequately (with on the order of 10,000 to 15,000 iterations), the worst case errors differed from one network to another by less than 1.0%. The worst case error for the testing-only welds was 7.9%. Although the errors in bead width estimates occasionally exceeded 20% (in less than 8% of the outputs), the width modeling accuracy is typically on the order of 10% or less. As Cook points out, the worst performance is obtained in the welding run in which the root width is recorded as larger than the crown width, which is an anomalous condition with respect to all other welds. The results, therefore, are within acceptable error levels and the tested network appears to constitute a workable model for the tested VPPAW parameters.

Process Control

The process control problem, as defined by NASA, was to determine the required equipment parameters (the IWP) to realize a set of desired output parameters (the DWP). To test the ability of a neural network to do this, a network was constructed to determine the torch standoff, forward current, reverse current, and travel speed for desired crown width and root width (refer to Fig. 7.11). Again, a single hidden layer was used, and the same thirteen welds were used as for the modeling experiment. For the parameter selecting network, however, the inputs were two (root width and crown width) and the outputs were four (torch standoff, forward current, reverse current, and travel speed). The same ten welds, as used for training the modeling network, were used for training this network. The network was trained through 15,000 iterations, resulting in sets of suggested torch standoff, forward current, reverse current, and travel speed selections. The network selected these parameters for each of the training welds, as well as for those which were not used in the training process.

To test the equipment parameter selection network shown in Fig. 7.11, its output was fed into the VPPA weld modeling network discussed previously (refer to Fig. 7.12). The desired crown and root

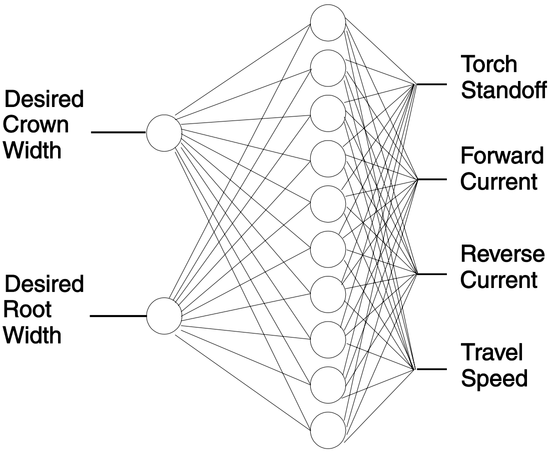


FIGURE 7.11 A neural network used for VPPA welding equipment parameter selection.

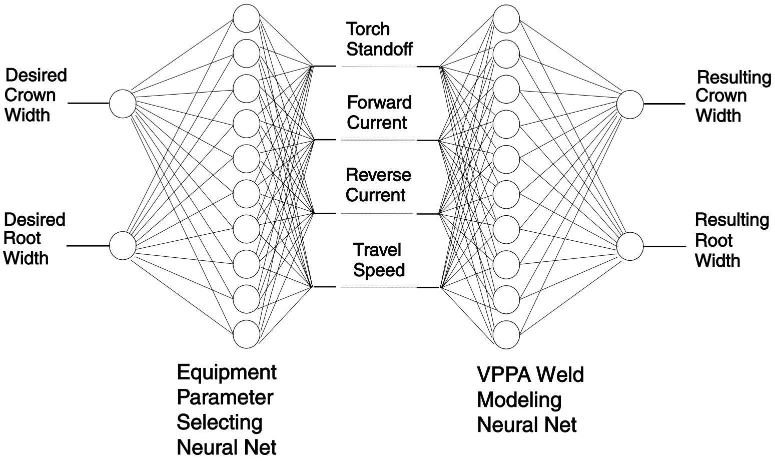


FIGURE 7.12 A parameter selection network determines the welding equipment parameters necessary to achieve desired bead geometry.

widths are specified and applied to the inputs of the equipment parameter selection network which, in turn, determines the suitable torch standoff, forward and reverse current, and travel speed. As a substitute for the actual welding process, the neural network VPPA weld model, discussed earlier, is fed with these equipment parameters and its crown and root widths are compared with the desired ones.

The deviations in the ultimate crown and root widths from the desired ones are mostly small. The worst case is again for the welding run in which the root width was recorded as larger than the crown width. Generally the welding data which was left out of the training of the parameter selector and the model does not exhibit noticeably worse performance than the training data. The conclusions from this experiment suggest that neural networks may be reliably used in selecting welding equipment parameters for the VPPAW process.

Weld Bead Profile Analysis and Control

A welding seam tracker, based on a laser scanner, was used at the NASA Marshall Space Flight Center for scanning VPPA weld profiles in near-real time. The seam tracking system is shown schematically in Fig. 7.13.

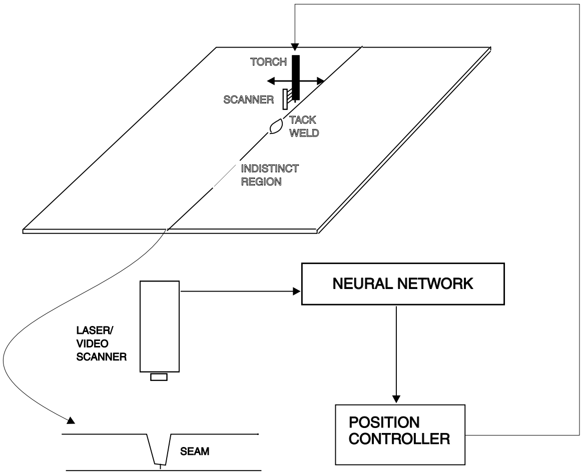


FIGURE 7.13 The seam tracking system.

The objective of the seam tracker is to monitor the location of the unwelded seam with respect to the moving welding torch, and to adjust the location of the torch with respect to the seam in real-time. In addition to seam tracking applications, the profile scanning system can be utilized to evaluate the quality of the VPPA weld. The quality of a VPPAW root pass can, to a large extent, be determined from the shape of its surface profile. Therefore, means for automatically detecting abnormal shapes, excessive undercuts, asymmetry of the weld profile, etc., are important.

Referring to Fig. 7.13, the laser scanner scans the seam in front of the moving torch and locates the seam with respect to the torch. Although the laser signal produces an adequate indication of this location most of the time, there are occasions when this method may become unreliable or fail. One situation is where isolated tack welding points along the welding seam cover the seam line, resulting in a laser signal that confuses the tracking system. Another cause for tracking errors occurs when the junction between the joined parts (e.g., for butt welds) is faint, or blends in with the parent metal surfaces, so that the seam apparently disappears over a section of the joint. Previously used data analysis algorithms for processing the output of the laser scanner sometimes became unreliable when the signal indicating the seam location became degraded in the manner discussed. The ability of the neural network to ignore minor disturbances made it an ideal candidate for this purpose.

A weld bead profile is obtained from the laser scanner where it is available as a list of coordinates. Typically, the list contains on the order of 80 coordinates, obtained for a fixed cross section of the weld profile. Each coordinate consists of the location of the measured point, as measured transversely across the bead width (y-axis value) and the height of the bead at that location (z-axis value). To locate, for example, the crown, undercuts and parent metal boundaries of the bead, the entire 80 data point heights (z-values) are fed into the network as distinctive inputs (refer to Fig. 7.14). Based on this given profile the network determines the locations of the desired bead profile parameters and presents these values as five separate outputs. Given the locations of the crown, undercuts, and bead boundaries, the crown height and undercut levels are easily determined by looking up the z-axis values at these locations. Deriving other properties of the bead, such as symmetry of the undercuts, both in terms of location and magnitude, is straightforward once the above information has been obtained.

Referring to Fig. 7.14, the height values of the digitized profiles were entered into the 80 inputs of the neural network, which was trained to locate the five features of the profile and present these locations at the outputs. Figure 7.15 illustrates a typical cross-sectional weld profile. The objective of the neural network was to find the locations of the weld crown, the left and right undercut, and the left and right weld boundaries. To the human observer, these features are relatively clear, with the possible exception of the weld boundaries. The locations of these features, as determined by the neural network, are indicated

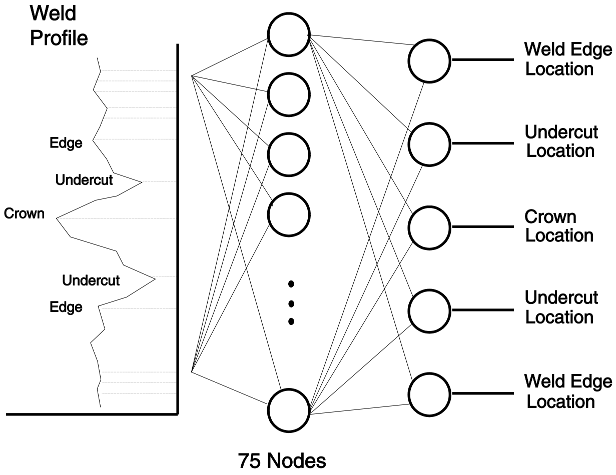


FIGURE 7.14 The neural network system used to locate the crown, undercuts, and weld boundaries from weld profile data.

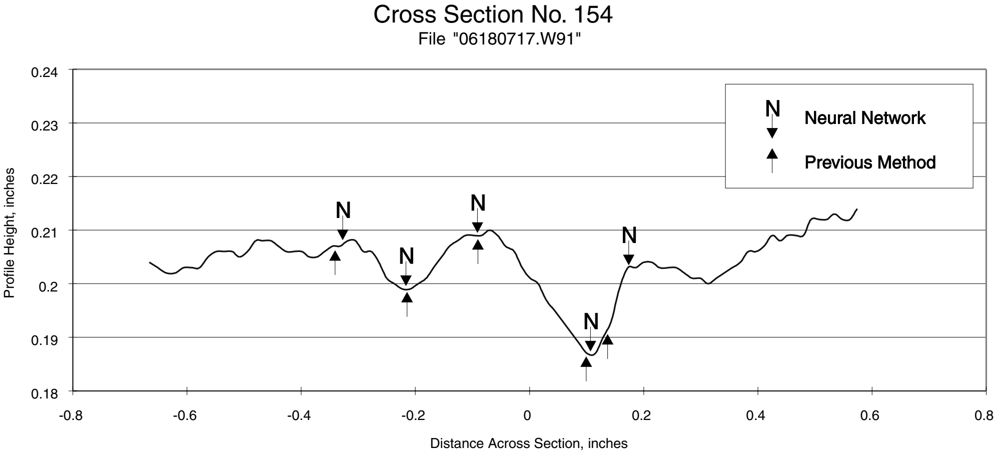


FIGURE 7.15 A scanned weld profile, with the weld crown, undercuts, and parent metal boundaries indicated by the proposed neural network and the previously used system.

on Fig. 7.15 with the arrows marked "*N* ". For comparison, the previously used data analysis algorithms located the same features as shown with the arrows pointing from below the profile trace. In this case the previous method failed in correctly identifying the right weld boundary, while the neural network succeeded. In most cases the previous system performed satisfactorily, while occasionally it gave incorrect results. The errors usually occurred in determining the weld boundaries, and these errors could be significant. For comparison, the neural network performed somewhat better, on the average, and the worst case errors were always much smaller than those of the previous system.

A total of 60 laser bead profiles were used to test and train the neural network used in the weld profile analysis. Of the 60 profiles taken, 30 were used for training and 30 were used for testing the trained network. Six of the testing profiles were sampled for error analysis, and the resulting errors were typically less than 1%.

7.8 Conclusion

A substantial amount of research is being conducted with the goal of developing an intelligent, fully automatic welding system. Many different intelligent schemes, including neural networks, fuzzy logic, neuro-fuzzy models, and rule-based expert systems, are being investigated. These intelligent methods have been applied (both statically and dynamically) to control the behavioral characteristics of the welding process in order to improve quality, reliability, and productivity in all industrial areas.

Considered as both a pro and a con, neural network control theory utilizes a more experimental and heuristic approach rather than a traditional adaptive control theorist approach to modeling control problems. The welding process has been traditionally difficult to analytically model due to the highly coupled, nonlinear nature of its system parameters. Despite these complexities, neural network learning techniques, like the back propagation algorithm, can model these traditionally difficult, nonlinear problems to a sufficiently high degree of accuracy [81]. Consequently, researchers in many different fields, including welding, have found success in employing artificial neural nets to previously unsolvable control problems.

Many traditional control theorists find fault with neural network solutions to parameter estimation due to its undisciplined, non-mathematical approach to modeling. Furthermore, even though a neural network can accurately model any function, the solution does not typically provide any physical insight into the understanding of the process. Nevertheless, unlike traditional adaptive control methods, neural networks are able to provide solutions to an otherwise unsolvable area of control problems.

Neural network systems have been shown to provide accurate off-line quality control based on the observations of the weld bead profile. In an attempt to eliminate the human operator from the feedback loop, neural networks have been successfully used to enact corrective actions on system variables in real-time. It has been found that neural networks are well suited to solving complex, highly coupled, nonlinear control problems inherent in the behavior of the fusion welding process.

References

1. Cook, G.E., Feedback and adaptive control in automated arc welding systems, *Metal Construction,* 13:9:551, 1981.
2. Cook, G.E., Robotic Arc Welding: Research in Sensory Feedback Control. *IEEE Transaction on Industrial Electronics*, IE-30:3:252, Aug 1983.
3. Cook, G.E., Andersen, K., Barnett, R.J., Feedback and Adaptive Control in Welding, *Recent Trends in Welding Science and Technology,* Eds.:S.A. David and J.M. Vitek, ASM International, Materials Park, OH, p. 891, 1990.
4. Cook, G.E., Andersen, K., Barnett, R.J., Welding and bonding, *The Electrical Engineering Handbook,* Ed.:R.C. Dorf, CRC Press, Inc., Boca Raton, FL, p. 2223, 1993.
5. Cook, G.E., Barnett, R.J., Andersen, K., Strauss, A.M., Weld Modeling and Control Using Artificial Neural Networks, *IEEE Transactions on Industry Applications,* 31:8:1484, 1995.
6. Andersen, K., Cook, G.E., Karsai, G., Ramaswamy K., Artificial Neural Networks Applied to Arc Welding Process Modeling and Control, *IEEE Transactions on Industry Applications,* 26:5:824, 1990.
7. Andersen, K., Cook, G.E., Ramaswamy, K., Karsai, G., A Novel Approach Toward Relationships Between Process Variables and Weld Geometry, *Recent Trends in Welding Science and Technology,* Eds.: S.A. David and J.M. Vitek, ASM International, Materials Park, OH, p. 997, 1990.
8. Andersen, K., Barnett, R.J., Springfield, J.F., Cook, G.E., Welding Technology: VPPAW Modeling and Control Using Artificial Neural Networks, SBIR Phase I Final Report, NASA Contract no. NAS8-38918, Aug, 1991.
9. Andersen K., *Studies and Implementation of Stationary Models of the Gas Tungsten Arc Welding Process,* M.S. Thesis, Vanderbilt Univ., Spring 1992.

10. Andersen, K., *Synchronous Weld Pool Oscillation for Monitoring and Control,* Ph.D. Dissertation, Vanderbilt Univ., Spring 1993.
11. Karsai, G., Andersen, K., Cook, G.E., Ramaswamy, K., Dynamic Modeling and Control of Nonlinear Processes Using Neural Network Techniques, Proceedings of the IEEE '89 Symp. Intelligent Control, Albany, NY, 280, Sept 1989.
12. Dornfeld, D.A., Tomizuka, M., Langari, G., Modeling and Adaptive Control of Arc Welding Processes, ASME Special Publication: Measurement and Control for Batch Manufacturing, Nov 1982.
13. Jackson, C.E., Shrubsall, A.E., Control of Penetration and Melting Ratio with Welding Technique, *Welding Journal,* 32:4:172s, 1953.
14. McGlone, J.C., Weld Bead Geometry Prediction—A Review, *Metal Construction,* 14:7:378, 1982.
15. Lin, M.L., Eagar, T.W., Influence of Surface Depression and Convection on Arc Weld Pool Geometry, *Transport Phenomena in Materials Processing,* ASME, p. 63, 1984.
16. Eagar, T.W., Oreper, G.M., Szekely, J., Convection in Arc Weld Pools, *Welding Journal,* 62:11:307s, Nov, 1983.
17. Lin, M.L., Eagar T.W., Influence of Arc Pressure on Weld Pool Geometry, *Welding Journal,* 64:6:163s, June, 1985.
18. Eagar, T.W., The Physics and Chemistry of Welding Processes, *Advances in Welding Science and Technology,* S.A. David, Metals Park, Ohio, ASM International, 1986.
19. Bates, B.E., Hardt, D.E., A Realtime Calibrated Thermal Model for Closed Loop Weld Bead Geometry Control, *Transactions of the ASME, Journal on Dynamic Systems, Measurement, and Control,* p. 25, March 1985.
20. Eraslan, A.H., Zacharia, T., Aidun, D.K., WELDER: A Computer Code for Simulating Fast-Transient, Three-Dimensional, Three-Phase, Flow, Temperature and Material Composition Conditions During Welding, Department of MIE, Clarkson University, Potsdam, NY, Report MIE-142, October 1986.
21. Zacharia, T., Eraslan, A.H., Aidun, D.K., Modeling of Autogenous Welding, *Welding Journal,* 67:3:53s, 1988.
22. Zacharia, T., Eraslan, A.H., Aidun, D.K., Modeling of Non-Autogenous Welding, *Welding Journal,* 67:1:18s, 1988.
23. Zacharia, T., David, S.A., Vitek, J.M., DebRoy, T., Weld Pool Development during GTA and Laser Beam Welding of Type 304 Stainless Steel Part 1-Theoretical Analysis, *Welding Journal,* 68:12:499s, 1989.
24. Zacharia, T., David, S.A., Vitek, J.M., DebRoy T., Weld Pool Development during GTA and Laser Beam Welding of Type 304 Stainless Steel Part 2-Experimental Correlation, *Welding Journal,* 68:12:499s, 1989.
25. Ohshima, K., Yabe, M., Akita, K., Kugai, K., Kubota, T., Yamane, S., Sensor Fusion Using Neural Network in the Robotic Welding, IEEE Industry Applications Conference-IAS 30th Annual Meeting, Orlando, p. 1764, 1995.
26. Yamane, S., Kaneko, Y., Kitahara, N., Ohshima, K., Yamamoto, M., Neural Network and Fuzzy Control of Weld Pool with Welding Robot, IEEE Industry Applications Conference-IAS 28th Annual Meeting, Toronto, p. 2175, 1993.
27. Messler, R.W., Jr., Jou, M., Li C.J., An Intelligent Control System for Resistance Spot Welding Using a Neural Network and Fuzzy Logic, IEEE Industry Applications Conference-IAS 30th Annual Meeting, Orlando, p.1757, 1995.
28. Richardson, R.W., Gutow, D.A., Rao, S.H., A Vision-Based System for Arc Weld Pool Size Control, *Measurement and Control for Batch Manufacturing,* Ed.: D.E. Hardt, ASME, New York, 1982.
29. Rokhlin, S.I., Guu, A.C., A Study of Arc Force, Pool Depression, and Weld Penetration During Gas Tungsten Arc Welding, *Welding Journal,* 72:8:381s, 1993.
30. Zhang, Y.M., Wu, L., Walcott, B.L., Chen, D.H., Determining Joint Penetration in GTAW with Vision Sensing of Weld Face Geometry, *Welding Journal,* 72:10:463s, 1993.

31. Zhang, Y.M., Kovacevic R., Wu L., Dynamic Modeling and Identification of Gas Tungsten Arc Welding Process for Weld Penetration Control, *Transactions of the ASME, Journal of Engineering for Industry,* 118(1):123, 1996.
32. Zhang, Y.M., Cao, Z.N., Kovacevic, R., Numerical Analysis of Weld Pool Penetration in GTA Welding, *the Proc. Inst. Mech.Engrs., Part B. Journal of Mechanical Engineering,* 210(c2):187, 1996.
33. Zhang, Y.M., Kovacevic, R., Li, L., Adaptive Control of Full Penetration in GTA Welding, *IEEE Transactions on Control Systems Technology,* 4:4:394, 1996.
34. Zhang, Y.M., Kovacevic, R., Wu, L., Dynamic Analysis and Identification of Gas Tungsten Arc Welding Process for Weld Penetration Control, *ASME Transactions, Journal of Engineering for Industry,* 118:1:123, 1996.
35. Kovacevic, R., Zhang, Y.M., Ruan S., Sensing and Control of Weld Pool Geometry for Automated GTA Welding, *Transactions of the ASME, Journal of Engineering for Industry,* 117:2:210, 1995.
36. Kovacevic, R., Zhang, Y.M., Real-Time Image Processing for Monitoring of Free Weld Pool Surface, *ASME Transactions, Journal of Engineering for Industry,* 118:4, 1996.
37. Kovacevic, R., Zhang, Y.M., Li, L. Monitoring of Weld Joint Penetration Based on Weld Pool Geometrical Appearance, *Welding Journal,* 317s, Oct 1996.
38. Kovacevic, R., Cao Z.N., Zhang Y.M., Role of Welding Parameters in Determining the Geometrical Appearance of Weld Pool, *ASME Transactions, Journal of Engineering Materials and Technology,* 118:3, 1996
39. Kovacevic, R., Zhang, Y.M, Apparatus and method for Measuring 3D Weld Pool Shape, United States Patent, #5,481,085, Jan 1996.
40. Kovacevic, R., Zhang, Y.M., Neurofuzzy Model-Based Weld Fusion State Estimation, *IEEE Control Systems,* 17:2:30, 1997.
41. Chiu S., Developing Commercial Applications of Intelligent Control, *IEEE Control Systems,* 17:2:94, 1997.
42. Brainmaker: Simulated Biological Intelligence, California Scientific Software, Nevada City, California, 1993.
43. Hunt, K.J., et. al., Neural Networks for Control Systems: A Survey, *Automatica,* 28:6:1083.
44. Zeng, X.M., Lucas, J., Fang M.T.C., Use of Neural Networks for Parameter Prediction and Quality Inspection in TIG Welding, *Transactions of the Institute of Measurement and Control,* 15:2:87,1993.
45. Huang, H.H., Wang, H.P., Tandem Neural Networks for Welding Process Control, *Journal of Systems Engineering,* 2:295, 1992.
46. Shirali, A.A., Mills, K.C., The Effect of Welding Parameters on Penetration in GTA Welds, *Welding Journal,* 72:7:347s, 1993.
47. Hardt, D.E., Garlow, D.A., Weinert, J.B., A Model of Full Penetration Arc-Welding for Control System Design, *Journal of Dynamic Systems, Measurement and Control,* 107:?, 1985.
48. Hardt, D.E., Measuring weld pool geometry from pool dynamics, *Modeling and Control Casting and Welding Processes,* Eds.: S. Kou and R. Mehrebian, The Metallurgical Society, Warrendale, PA, 1986.
49. Zackenhouse, M., Hardt, D.E., Weld pool impedance identification for size measurement and control, *Transactions of the ASME, Journal of Dynamic Systems, Measurement, and Control,* 105(3) 179, 1983.
50. Suzuki, A., Hardt, D.E., Valvani, L., Application of Adaptive Control Theory to On-Line GTA Weld Geometry Regulation, *Journal of Dynamic Systems Measurement and Control,* 113:93, 1991.
51. Barnett, R.J., *Sensor Development for Multi-Parameter Control of Gas Tungsten Arc Welding,* Ph.D. Dissertation, Vanderbilt Univ., Spring 1993.
52. Connor, L.P., ed. *Welding Handbook,* vol. 2, *Welding Processes,* American Welding Society, Miami, FL, 1991.
53. Barto, A.G., Connectionist learning for control: and overview, *Neural Networks for Control,* Eds.: W.T. Miller, III, R.S. Sutton, P.J. Werbos, The MIT Press, Cambridge, 1990.

54. Rosenblatt, F., The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, *Psychological Review,* 65:6, 1958.
55. McCulloch, C.A., Pitts W., A Logical Calculus of the Ideas of Imminent in Nervous Activity, *Bull. Math. Biophys.,* 5:115, 1943.
56. Widrow, B., Hoff, M.E., Adaptive Switching Circuits, *1960 IRE WESCON Conv. Rec.,* pt. 4, p. 96, Aug, 1960.
57. Rumelhart, D.E, McClelland, J.L, *Parallel Distributed Processing,* vols. 1 & 2, MIT Press Cambridge,1986.
58. Lippmann, R.P., An introduction to computing with neural nets, *IEEE ASSP Magazine,* 4, 1987.
59. Lippmann, R.P., Gold, B., Malpass, M.L., A Comparison of Hamming and Hopfield Neural Nets for Pattern Classification, MIT Lincoln Laboratory Technical Report, TR-769.
60. Hopfield, J.J., Neural Networks and Physical Systems with Emergant Collectice Computational Abilities, *Proceedings of the National Academy of Sciences,* USA, Col. 79, 2554, April 1982.
61. Macvicar-Whelan, P.J., Fuzzy sets for man-machine interaction, *Int. J. Man-Mach. Studies,* 8:687, 1976.
62. Procyk, T.J., Mamdani, E.H., A linguistic self-organizing process controller, *Automatica,* 15:15, 1979.
63. Brase, M., Rutherford, D.A., Theoretical and linquistic aspects of the fuzzy logic controller, *Automatica,* 13:553, 1979.
64. Eassa, Hussam El-Dean, E.H., Electric Arc Welding Control with a High Performance Power Source, Ph.D. Dissertation, Vanderbilt Univ., 1983.
65. Rosenthal, D., Mathematical Theory of Heat Distribution During Welding and Cutting, *Welding Journal,* 20:5:220s, 1941.
66. Nunes, Jr. A.C., An Extended Rosenthal Weld Model, *Welding Journal,* p. 165, June 1983.
67. Nunes, Jr. A.C., Bayless E.O., Jones, C.S., III, Munafo, P.M., Biddle, A.P., Wilson, W.A., Variable Polarity Plasma Arc Welding on the Space Shuttle External Tank, *Welding Journal,* 63:9:27, 1984.
68. Tomsic, M., Barhorst, S., Keyhole Plasma Arc Welding of Aluminum with Variable Polarity Power, *Welding Journal,* 63:2:25, 1984.
69. Tsai, N., Heat Distribution and Weld Bead Geometry in Arc Welding, Ph.D. Thesis, M.I.T., April 1983.
70. Tsai, C.L., Dai, W.L., Dickinson D.W., Papritan J.C., Analysis and Development of a Real-Time Control Methodology in Resistance Spot Welding, *Welding Journal,* 20:5:220s, 1991.
71. Tsai, M.C., Kou, S., Electromagnetic-Force-Induced Convection in Weld Pools with a Free Surface, *Welding Journal,* 69:6:241, 1990.
72. Kim, J.W., Na, S.J., A Self-Organized Fuzzy Control Approach to Arc Sensor for Weld Joint Tracking in Gas Metal Arc Welding of Butt Joint, *Welding Journal,* 72:60s, 1993.
73. Reeves, R.E., Manley T.D., Potter A., Ford, D.R., Expert System Technology – an Avenue to an Intelligent Weld Process Control System, *Welding Journal*, 67: 33, 1988.
74. Sugitani, Y., Nishi Y., Sato T., Intelligent Robot Controls Penetration and Bead Height, *Welding Journal,* 69:31, 1990.
75. Jang, J.S.R. , Sun C.T., Neuro-Fuzzy Logic Modeling and Control, *Proceedings of the IEEE,* 83:3:378, 1995.
76. Du, R., Elbestawi, M.A., Wu, S.M., Automated Monitoring of Manufacturing Processes, Part 1: Monitoring Methods, *ASME Journal of Engineering for Industry,* 117:2:121, 1995.
77. Sharaf, A.M., Lie, T.T., Neuro-Fuzzy Hybrid Power System Stabilizer, *Electric Power Systems Research,* 30:1:17, 1994.
78. Song, J.B., Hardt, D.E., Closed-Loop Control of Weld Pool Depth Using a Thermally Based Depth Estimator, *Welding Journal,* 72:10:471s, 1993.
79. Mesina, O.S., Langari, R., Neuro-Fuzzy System for Tool Condition Monitoring in Metal Cutting, *Dynamic Systems and Control,* 2, p. 931, 1994 ASME International Mechanical Engineering Congress, ASME, 1994.

80. Takagi, T., Sugeno, M., Fuzzy Identification of Systems and Its Applications to Modeling and Control, *IEEE Transactions on Systems, Man, and Cybernetics,* 15:1:116,1985.
81. Hornik, K., Stinchcombe, M., White, H., Multilayer Feedforward Networks are Universal Approximators, *Neural Networks,* 2:p. 359.
82. Barto, A.G., Connectionist Learning for Control, *Neural Networks for Control* (W.T. Miller, R.S. Sutton, and P.J. Webos, eds.), p. 9, MIT Press, Cambridge, MA, 1990.
83. Eberhart, R.C., Standardization of Neural Network Terminology, *IEEE Transactions Neural Networks,* 1:2:244, 1990.
84. Willis, M.J., Montague, G.A., di Massimo, C., Tham, M.T., Morris, A.J., Artificial Neural Networks in Process Estimation and Control, *Automatica,* 28:6:1181, 1992.
85. Frohlinghaus, T., Weichert, A., Rujan, P., Hierarchical Neural Networks for Time-Series Analysis and Control, *Network: Computation in Neural System,* 5:101, 1994.